AD-A052 631 TEXAS TECH UNIV LUBBOCK F/G 5/9
OPTIMIZATION TECHNIQUES FOR AUTOMATED ADAPTIVE TRAINING SYSTEMS--ETC(U)
JUL 77 D C CHATFIELD , C F GIDCUMB N61339-77-M-0575
UNCLASSIFIED NAVTRAEQUIPC-77-M-0575 NL

| OF |

AD A052631

END
DATE
FILMED
5-78
DDC

AD A052631

AD No. ____

DDC FILE COPY

(12)

Technical Report NAVTRAEQUIPCEN 77-M-0575

OPTIMIZATION TECHNIQUES FOR AUTOMATED ADAPTIVE TRAINING SYSTEMS

Texas Tech University
Lubbock, Texas 79409

July 1977

DoD Distribution Statement

Approved for public release; distribution unlimited.

Naval Personnel Research and Development Center
San Diego, California

NAVAL TRAINING EQUIPMENT CENTER
ORLANDO, FLORIDA 32813

NAVTRAEQUIPCEN 77-M-0575

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
NAVTRAEQUIPCEN 77-M-0575

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
Optimization Techniques for Automated Adaptive Training Systems.

5. TYPE OF REPORT & PERIOD COVERED
Final Report, Jan [redacted]-Jul [redacted]77

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Douglas C. Chatfield, Charles F. Gidcumb

8. CONTRACT OR GRANT NUMBER(s)
N61339-77-M-0575

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Texas Tech University
Lubbock, Texas 79409

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
4754-4P1

11. CONTROLLING OFFICE NAME AND ADDRESS
Human Factors Laboratory
Naval Training Equipment Center
Orlando, Florida 32813

12. REPORT DATE
Jul [redacted]77

13. NUMBER OF PAGES
58

62 p.

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
NPRDC
Naval Personnel Research & Development Center
San Diego, California

15. SECURITY CLASS. (of this report)
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Approved for public release; Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Automated Adaptive Training
Automated Performance Measurement
Optimization Techniques
Automated Training Systems
Learning Models
Optimal Control Theory
Flight Training
Cognition

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
As adaptive training systems are developed, the big problem encountered is the development of an adaptive logic. Current systems develop their branching schemes without direct usage of learning models. The literature presently contains several new techniques for optimizing instruction. These techniques make use of current learning models by which to make trial-by-trial estimates of the student's progress in learning. This information is then used to solve a set of equations which would select an optimal instructional alternative. Optimal decisions are made by selecting the instructional

over

DD FORM 1473, 1 JAN 73 — EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

405 703

alternative for which the largest marginal gain in learning is predicted. The present task was to review the techniques available which present the greatest feasibility for applications in the developing training systems. The various optimization techniques selected were presented in their most general form so that the variety of their applications might be apparent. It was concluded that the optimization techniques reviewed were quite feasible and have many powerful options to offer.

SUMMARY

INTRODUCTION

The Navy is presently engaged in the development of automated adaptive training systems. For this reason, new technological demands are being made on the educational researcher. One such demand concerns the techniques for the development of an adaptive logic to be used in automated training systems.

The introduction presents a general scheme by which adaptive logics may be developed. This scheme involves a principle referred to as "optimization". Optimization refers generally to the idea that, in using a learning model to predict current stages of learning, one optimizes learning while minimizing costs, by appropriate choices of instructional alternatives. Concepts within the principle of optimization are exemplified through three simple techniques which are based on three learning models. It is further demonstrated that the techniques to be proposed could be implemented in the developing automated training systems. Some of the decision-making functions within an existing adaptive system may be taken over by one of the optimization techniques.

PROBLEM

As adaptive training systems are developed, the big problem encountered is the development of an adaptive logic. Current systems develop their branching schemes without direct use of learning models. Thus the first question becomes: What optimization techniques are presently available in the literature which could determine such things as task-selection and branching logic? Secondly, how do the optimization techniques compare with current non-theoretical methods? Lastly, how feasible are the techniques in terms of what development would still be required before implementation?

GENERIC TECHNIQUES FOR OPTIMIZATION

For comparison purposes, the development of the task selection portion of a current automated system under development is reviewed. In this case it is pointed out that the system does not base its task selections on estimated gains in learning but rather on predicted performances. The ramifications of this are explored and compared with the other techniques.

The different optimization techniques are presented in their most general form so that the variety of their applications might be apparent. The particular techniques selected are the ones judged as most feasible for dealing with the adaptive logic in an automated system. The organization is such that the techniques are presented in three categories. The first category covers the situation wherein the units-of-presentation (tasks, exercises, problems, etc.) are the objectives of the training process

themselves, and are designated as the units-of-acquisition. The other two categories represent techniques which assume that the units-of-acquisition are few in number, relative to the units-of-presentation. Here the objectives of the training involve the acquisition of underlying skills or concepts, and not necessarily the exercises or tasks which are presented. These last two categories of techniques are further differentiated by whether or not the acquisition of these underlying skills is assumed to be continuous or discontinuous.

CONCLUSIONS

It was concluded that the optimization techniques reviewed are quite feasible and present the designers of the adaptive systems with many more powerful options than they presently have. Suggestions were given as to the development requirements for implementation in the short-run. Suggestions were also given concerning the future directions of longer term development and the benefits which could be expected.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

SECTION I

INTRODUCTION

Adaptive training got its start during the late 1950's in the design of tracking devices. As the trainee improved his performance, the device adapted automatically to make the task more difficult. The automatic change produced a constant rate of performance from the trainee, but indicated improvement since the task was more difficult.

Kelley (1969), who was among the first to use adaptive training in simulators, defined adaptive training as the varying of difficulty of the to-be-learned task as a function of the performance of the trainee. A more current definition by Atkinson (1976) includes a first part essentially the same as that of Kelley (1969). Atkinson goes on to add that the program of the adaptive system itself adapts as the number of students using the system increases and their performance records identify possible improvements in the initial instructional strategies.

Before the implementation of adaptive training devices, most automated training devices followed a preprogrammed sequence without regard for the student's performance. Such a system is termed an open-loop or response-insensitive system since the control is preprogrammed. Adaptive training follows a closed loop in which the trainee's performance is considered in the generation of the next learning trial. In other words, the trainee's performance feedback to the controller affects problem generation. Poor performance results in easier tasks while better performance leads to more difficult tasks or problems. Such a feedback system is depicted schematically in Figure 1.

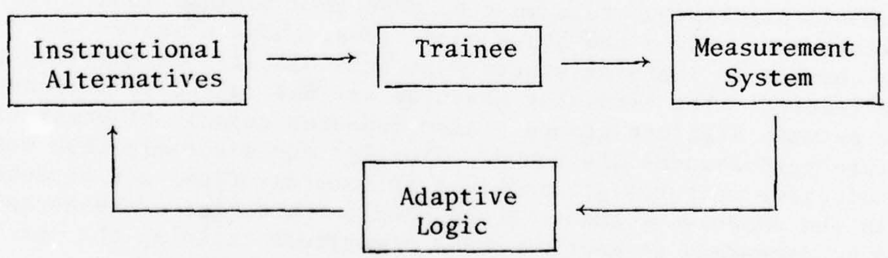

Figure 1. Basic Configuration of an Adaptive System

The three required items of an adaptive training system are indicated. The first is a set of instructional alternatives e.g., the selection of problems or tasks from a pool of problems. The curriculum is broken into individual tasks by an expert or by instructional personnel familiar with the overall training objectives. The adaptive variable would be any adjustable

feature of the assigned tasks or problems which can be modified as a result of an individual student's performance. Many things can become adaptive variables such as pacing, mode of presentation, number of variables with which the trainee must contend, or the task selection itself.

The second requirement is that there exist a measurement system which scores the trainee's performance and feeds the information to the adaptive logic. The measure used must be reliable and pertinent to the system since the adaptive variable is changed on the basis of the measure. In many cases the measure is averaged over time or over trials, but discrete measures may also be used such as the correctness of the student's response.

The third required part of an adaptive system is the adaptive logic. The adaptive logic is a set of decision rules specified to determine how the adaptive variable is tied to the performance measure. The logic can be specified as a mathematical relationship or a specific adjustment rule. An adjustment rule as an example could require the system to increment to the next most difficult level after a fixed number of consecutive correct responses.

Current developments in the area of instructional alternatives are still somewhat limited in scope. Generally the allowable alternatives are specific to the curriculum to be learned. The content of the alternatives is still developed by the expert or instructional team.

Vruels and Goldstein (1974) offered a process to improve the selection of measurement. Initially a possible set of measures must be developed. Knowledge of the task can be used to generate possible measures or they can be obtained from the literature. In order to evaluate the measures the raw data parameters must be determined and necessary transforms established. Along with this an unambiguous rule must be developed to know when to start and stop measurement. Given the above conditions, those measures which are sensitive to changes in learning states must be isolated. Next, redundant measures are removed. The remaining measures are now sorted into those that discriminate between distinct groups. Also measures capable of predicting possible future performances are found. Finally, the discriminative measures and predictive measures are combined to form the final set of measures to be used in the adaptive system. These complex batteries of measures are suggested to be necessary to meet the needs of systems training the more complex tasks found today according to Conway and Norman (1974).

While the earlier adaptive systems were fairly simple, having only one controller, newer systems utilizing multiple controllers in a hierarchy are now being advocated. For example, the lower controller would implement one possible strategy from a class of possible predefined control strategies. The higher controller would decide which strategy is to be implemented. In performing this function the higher level controller assumes some of the functions traditionally performed by an instructor; i.e., the prescribing of predefined units of training.

Conway and Norman (1974) are advocating such higher order adaptive systems which would be self-organizing. Such a self-organizing system would take into account the specific learning style of the trainee when prescribing the specific strategy. In doing so, the system forces the instructor into the role of providing the instructional materials while the system makes decisions regarding the match of trainee and learning requirements.

Conway and Norman (1974) go on to list the qualities required of such a higher order system. First the system must be capable of making policy level and instructional decisions. Secondly, the system must be able to collect data on system and trainee performance toward the goal of learning how to train. Along this vein, the system must have the sensitivity to identify different learning styles with flexibility to organize training requirements around those styles. This self-modifying system would also allow the trainee to participate in the process of strategy and item selection. Finally, the flexibility to handle a wide spectrum of learning tasks from simple information training to complex psychomotor skills is required.

As the development and implementation of higher order adaptive systems progresses, two questions are being encountered. The first is in the development of the adaptive logic. More specifically, what are the objectives by which the system will make its adaptive decisions? Obviously the long-range objectives are the acquisition of the skills or informational content. But the general long-range objectives do not indicate whether a student who has just completed exercise (a) successfully, should be branched to exercise (b) or exercise (c). Secondly, if the system is to modify itself, on what basis is it to be programmed to do such.

In regard to the first question, there has been some recent progress on the development of adaptive logics within the realm of automated instruction. Basically, learning models are used to describe the student on a trial-by-trial basis so that the adaptive logic can base its decisions on the hypothesized learning state of the individual. Thus the grand objective can be broken down into local trial-by-trial objectives.

The learning models themselves, have been specified in a formalized mathematical form. Together with a formalized set of instructional alternatives, optimal control theory (see Howard, 1960) has been used to optimize mathematical functions representing the student's state of learning, with respect to the instructional alternatives in question. Herein lies an appropriate mechanism by which the adaptive logic of a higher order system could be devised.

This paper presents a review of some of the more promising techniques that have been developed. But before presenting the relatively complex schemes, it would be well to take a relatively simple example by which to develop a set of definitions, and to establish its context within the framework of higher-order adaptive training. The example is from Atkinson and Paulson (1972) and exemplifies the applied qualities of the techniques. To begin, assume that a portion of a designated curriculum could be defined

as a large set of independent problems representing separate skills, associations, or concepts to be acquired. Of this large set of say N problems, only M of the problems can be given at any one training session at the computer terminal (wherein $M < N$). We will assume first that every trainee has a fixed number of days to master as many of the N problems (and thus the corresponding N skills) as possible; second, each of the problems or skills represented are of equal importance or benefit; and third, each problem takes an equal amount of time (and cost) to present.

At this point, it would be good to point out, that though we assumed all skills to be of equal benefit, and that all problems were of equal cost to present, these are simplifying assumptions that may be relaxed. To relax these assumptions however requires that we be able to specify cost and benefit. This specification will later be called the cost/benefit structure.

LINEAR OPERATOR MODEL

It would seem reasonable to seek to maximize the proportion of the N problems (or skills) mastered, within the constraints of the fixed number of sessions, and the fixed number (M) of items presented per session. Within the situation outlined, it can be seen that the instructional decisions to be made reduce to simply a choice of which of the N problems are to be selected for presentation at each session for each trainee. In order for the computer to make optimal choices, a model of the learner is needed. Again for illustration, two simplified types of learning models are used. The first, termed the incremental model, simply assumes that if a particular problem is presented, a small increment in the corresponding skill results. Let q_n represent the probability of an error being committed on the particular problem in question on the nth presentation. Now on the next presentation (presentation n+1) the probability of an error will be assumed to be reduced by the fraction a, say $q_{n+1} = aq_n$ where $0 < a < 1.0$. This is the linear operator learning model of Bush and Sternberg (1959). It represents a class of learning situations in which learning is seen as a gradual process. Now if $q_n = aq_{n-1}$ then the probability of an error on trial n_1 can be expressed as $q_n = a^{n-1}q_1$ where q represents the probability of an error on the first presentation, before any training has taken place.

Now in order to make a decision as whether or not to present a problem in a particular session, one would need to look at the marginal gain for presenting the problem. Say that the problem's current error probability is $q_n = a^{n-1}q_1$. If the problem is not presented, the error probability remains unchanged i.e., q_n. But if it is presented then the new error probability is $q_{n+1} = aq_n = a^n q_1$. Now the marginal difference between presenting vs. not presenting the problem is:

$$q_n - q_{n+1} = a^{n-1}q_1 - a^n q_1$$
$$= (1-a)a^{n-1}q_1 \qquad (1)$$

Note that in Eq. (1) the marginal difference is a function of n, the number of past presentations of the problem. Thus the most rational approach in

selecting problems for the next session for a trainee would be to select the ones that have been presented the least, i.e., the ones with the smallest n's. Thus the computer would be programmed to simply keep a record on each trainee and each problem from the pool of N problems. At the beginning of each session the computer would simply select the problems having been presented the least, until M problems are selected. This procedure can be shown to be the "optimal" strategy in the mathematical sense within the constraints given, and the assumption of this simple linear model.

Figure 2 represents the general organizational form of an existing training system with the addition of an item selection component. The figure should allow us to point out the way in which an optimization technique could mesh with an existing training system; and at the same time it illustrates some optimization concepts which will be needed later.

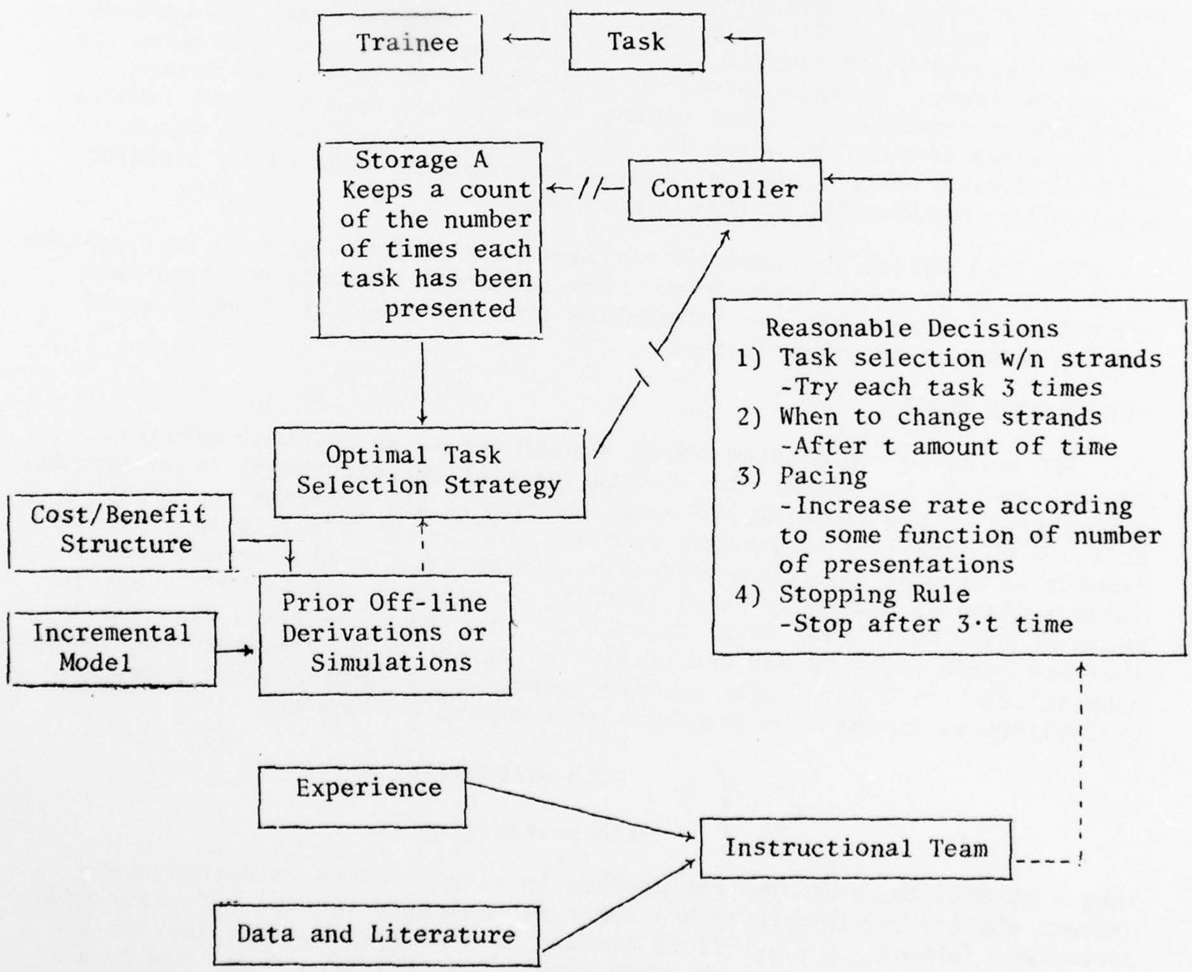

Figure 2. Open Loop (Response Insensitive) System

It will first be assumed that a particular system already exists and is represented in a highly simplified form by: the trainee, task, controller, and reasonable decision components. Concerning the last component, the term "reasonable decision" is used to denote the normal decisions that an instructional team must make in the design of an automated training system. The word "reasonable" is used to differentiate these solutions from what is being referred to as "optimal" solutions. "Reasonable" simply represents the idea that though the decisions may be intuitively plausible and compelling, they may be suboptimal. Typically, large numbers of instructional decisions must be made in the design of a training system and it is not our current intent to find optimal solutions for all of them. However, optimal solutions may be substituted for reasonable solutions whenever possible.

The system components behind the broken arrows denotes an alternate optimal solution which may be substituted for decision number 1. It represents the solution we have just outlined. Here the system stores the frequency with which each item has been presented and then uses the optimal strategy, previously discussed, by selecting the items which have been presented the least. At this point, it is essential to notice two things: first, the determination of the optimal strategy was done off-line (denoted by the dashed arrow); and secondly, this off-line determination, whether by analytic derivation or computer simulation, requires that both a learning model and a cost/benefit structure be specified.

The task selection component of Figure 2 represents an open loop system in that it is response insensitive. The selection strategy proposed does not make use of any response information from the student: thus it would not represent an adaptive system.

ALL-OR-NONE MODEL

The solution represented above, though termed an "optimal solution", is optimal only to the extent that the incremental learning model is an adequate description of the learning process. However if the problems represent skills with cognitive components then acquisition may actually be a discontinuous or even an all-or-none process. As an example of a second model, consider a class of models in which learning occurs on a single trial, but remains at some base level until then. Let c be the probability that the learning takes place on any one trial, and again let q_1 represent the error probability for a particular problem before any presentations. Then the probability of an error on presentation n may be written as:

$$q_n = \begin{cases} q_{n-1} \text{ with probability } (1-c) \\ 0 \text{ with probability } c. \end{cases} \tag{2}$$

Thus with this description, the problem is either in the unlearned state wherein the error probability has remained unchanged for all n-1 prior presentations (thus $q_{n-1} = q_1$) or is already in the learned state (having had made the transition on some previous trial) and thus $q_{n-1} = 0$. Now on a particular trial, if a problem ends in an error it could be reasoned that it was surely in the unlearned state. However, the converse is not true. If a

problem ends in a correct response, one possibility is that it was in the learned state; a second possibility is that it was in the unlearned state and was correct by chance (i.e., $1 - q_1$). Now the more consecutive successes in a row for a particular item, the greater the likelihood of it being in the learned state. This is important because it can be shown (see Calfee, 1970) that the optimal strategy for the all-or-none model in our exemplar situation, is to select for presentation those items having the greatest likelihood of being in the unlearned state. Thus the computer would keep error-success records on each problem and each trainee. At the beginning of each session, the computer would begin by selecting problems with zero successes since the last error, then those with one success since the last error and so on until a full set of M problems have been selected.

Figure 3 shows how the problem selection solution based on the all-or-none model might be implemented in the existing training system. It will be noted that the requirement for storage of information is simply a count of the number of successes since the last error. Also, it will be noticed that the selection strategy was again derived off-line, as it was in the previous example. Lastly, it should be pointed out that the solution based on the all-or-none model is a response sensitive system.

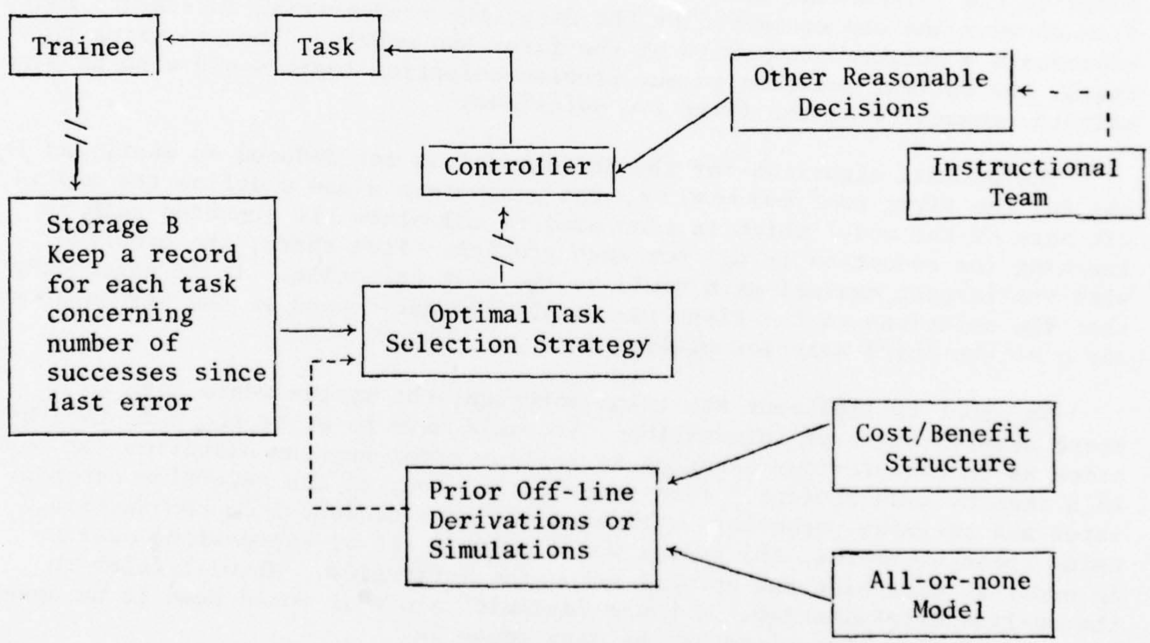

Figure 3. Response Sensitive System.

Atkinson and Paulson (1972) report a study by Lorton (1972) which compared the two different selection strategies in a computer-assisted program for elementary school children. Herein the problems consisted of words which the children were required to learn to spell. The results were that the selection strategies derived from the all-or-none model produced superior performance on both an initial post-test and delayed post-test. Thus in the spelling tasks with children, one could infer that the all-or-none model was a better description of the trainee.

RANDOM-TRIALS INCREMENTS MODEL

A third model which encompasses the advantages of both the all-or-none, and the incremental models is the random-trials increments model proposed by Norman (1964). The model, simply stated, can be represented by the difference equation

$$q_n = \begin{cases} q_{n-1} & \text{with probability } (1-c) \\ aq_{n-1} & \text{with probability } c, \end{cases} \tag{3}$$

where the parameters q_n, a, and c are the same as in Eqs. (1) and (2). It will be noted that if $a = 0$ and $0 < C < 1$, the model reduces to the all-or-none model presented in Eq. (2). But if $0 < a < 1$ and $c = 1$, the model reduces to the incremental model. Thus, if the parameters a and c are left free to vary and are estimated by the data, the random-trial increments model represents a weighted composite of the first two models. Thus in using this model, the optimal solution to our problem-selection logic would also be some weighted composite of the first two solutions.

The optimal algorithm for the third model is not deduced as easily as it was for the first two. Basically, the parameters a and c define the explicit form of the model which is then used to calculate the expected gain in learning (or reduction in q_n) for each problem. From there, the problem with the largest maximal gain would be the next selection. It is noteworthy that the solutions to the first two models did not depend on the values of a and c as the third solution would.

In order to implement the third solution, the system would have to store several items of information. It would have to store the same information as in the previous examples as well as error-success histories for each item by each student. From this information, it can determine error rates and on-going parameter estimates for each student-item combination. Using these estimates, the system would solve a set of expressions on-line in order to determine the optimal selection strategies. We will refer to the on-line determinations as being "dynamic" since it would need to be done during real-time operations as the data comes in.

Figure 4 illustrates how the system configuration might look. As can be seen, some form of psychometric information might be used for the initial parameter estimates of a and c. From there, the estimates could be updated from the information obtained during actual training operations. The ongoing estimates would be used on-line to determine optimal problem selection

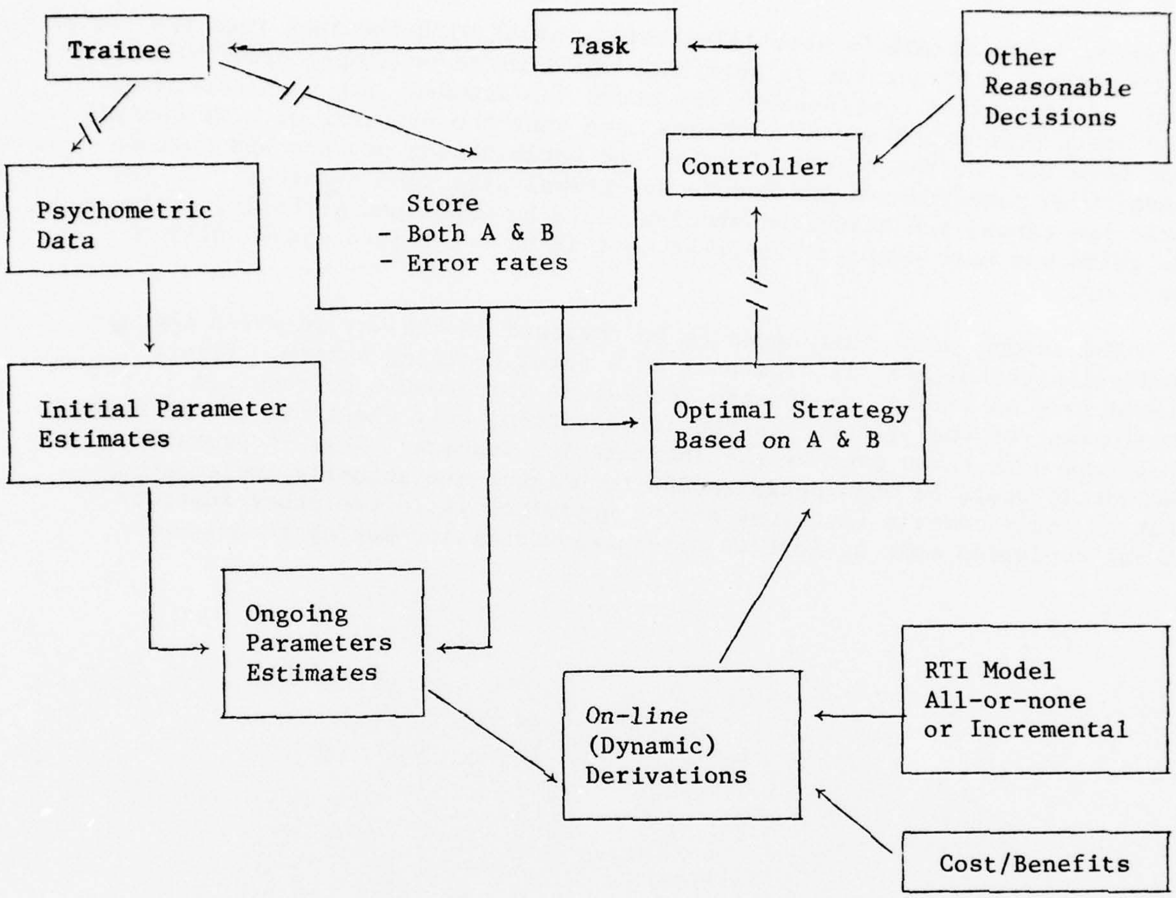

Figure 4. Adaptive System

strategies for each student. The system would be self-modifying to the extent that it accumulates parametric information on each problem as it gains experience from additional students. Atkinson & Paulson report a study using a parameter-dependent adaptive system similar to the one described here. They found that the overall performance of the system made gains with the successive groups of students. The estimates of item difficulties as represented in the parameters were crude at first, but improved and stabilized with succeeding groups of students. The techniques by which a system such as this gains stable parameter estimates will be taken up later.

APPLICATION

Before leaving the present sections, it would be good to point out the specific points which the three examples described were to illustrate. The first point was simply the general form of the optimization techniques. Specifically that one must have a model of the learning process, a specific set of instructional alternatives, and a cost/benefit structure. Three

different learning models were illustrated, along with the idea that the set of instructional alternatives in this case was limited to simple problem selection. A simplified cost/benefit structure was assumed, i.e., the presentation costs were equal across problems and that the problems were of equivalent benefit. So the general form of the optimization process was to take these three requirements and deduce an optimal selection strategy. In the first two cases, the deduction process could be explained verbally, while the third was more complex requiring that it be determined dynamically on the computer.

The second point that needs to be reviewed is the way in which the optimization techniques can be a part of a total training system. Figure 2 listed four so-called "reasonable" decisions which would be required in the development of the system. In the illustration, only the first of the four decisions were taken over by the optimization process. This is probably the way it would be when optimization techniques are actually implemented. That is, only certain functions may be optimized while the other instructional decisions must be left to alternative decision-making techniques.

SECTION II

PROBLEM

Currently there is considerable effort being invested in the development of self-organizing automated adaptive systems. It is the intent of these systems to make use of the extensive research on performance measurement in order to formulate proscriptive individualized training for students in such areas as flight training. The basic question which confronts the developers at this point concerns the adaptive logic in the system. On what basis are the performance measures to be used for individualized instruction, and on what basis is the system to modify itself?

There are several optimization techniques by which adaptive logics and instructional decisions can be derived from assumed learning models. Some of these techniques are suitable for implementation in current adaptive training systems. Thus the problem becomes one of identifying and reviewing those techniques most suitable. The techniques need to be examined in terms of the type of learning models assumed, and the type of instructional materials to which they would apply. They would need to be compared with each other in terms of instructional alternatives which they may handle and the specific functions utilized in the optimization process. It would also be helpful if they were compared to current systems under development. A final need is to obtain recommendations as to what would be necessary for implementation of any of the optimization techniques. The sections to follow will address these needs.

SECTION III

GENERIC TECHNIQUES FOR OPTIMIZATION

In this section, the optimization techniques judged most feasible are presented. The techniques center around that portion of the adaptive logic concerned with the selection of problems, tasks, or exercises. It was felt that task selection represented the most central and pervasive problem in current adaptive training developments. The general rubric of task selection would encompass several other problem categories. The determination of a branching logic would of course be a special case of task selection wherein the selection rules result in a complex form. Subsumed under the concept of branching logic would be the current content areas of diagnostics and remediation. Furthermore, the constant change in a continuous adaptive variable representing task difficulty of the type described by Kelley (1969) could also be viewed as a task selection problem.

It should also be pointed out that an attempt is made to review the techniques in their most general form. The learning models assumed by the techniques are in a form such that they represent a class of models. Some of the results deduced from the models are of such general form that they are not dependent upon the specific form of the model. Other results however may be specific and must be determined at the time of actual implementation.

The optimization techniques reviewed are grouped by the type of learning situation to which they might apply. The learning situations are grouped by a principle we would like to refer to as the "unit-of-acquisition". In developing this principle one would like to point out specifically what it is that is mastered in different situations. For example, in a paired-associate task, the unit-of-acquisition would be the individual associations which link each pair. This is to be contrasted with a second principle referred to as the "unit-of-presentation". In this example the unit-of-presentation is considered the stimulus-pair. Thus in paired-associate learning there is a one-to-one correspondence between the units-of-acquisition and the units-of-presentation, implying that the objectives of training would be the mastery of the individual items or tasks themselves.

Where the curriculum is composed of conceptual material, it is the underlying concepts that are the units-of-acquisition. Here, several items or tasks may be instances of a single concept. This implies a many-to-few mapping of the units-of-presentation to the units-of-acquisition. In this case the objectives of training would be the mastery of the hypothetical concepts.

A third case to consider is the situation wherein the units-of-acquisition represent continuous psychomotor skills. The units-of-presentation would be in the form of exercises. As in the conceptual case, there would be a many-to-few mapping between the units-of-presentation and the units-of-acquisition. Here again the objectives of training would be the inferred mastery of the underlying skills and not the individual exercises. As in the conceptual case, if the system has ascertained that a skill is acquired, some of its corresponding exercises may even be skipped. The difference between the conceptual case and the psychomotor example is that in the former, the learning models tend to represent learning progress in discrete steps or states, whereas the latter may represent learning increments as approaching infinitely small units.

The remainder of this section is organized by the ultimate objectives of training in terms of the units-of-acquisition. Thus the optimization techniques are grouped as to whether the units-of-acquisition are the individual tasks, the underlying concepts, or the underlying psychomotor skills. A further discussion of this conceptualization is presented in the Appendix.

Before reviewing the techniques themselves, it will be recalled from the introduction that it should be feasible to utilize the optimization techniques within the confines of an existing training system. That is, an optimization technique might be a viable replacement for one of the many so-called "reasonable" solutions. Thus, it would be good to review first an exemplar task-selection strategy in a current automated adaptive training system. This would facilitate determining the ultimate feasibility of substituting one of the optimal techniques for an existing technique.

TASK-SELECTION IN A CURRENT SYSTEM

It was suggested (Norman, 1977) that a typical program of immediate interest to NTEC would be the Higher-Order Partially Self-Organizing (HOPSO) adaptive training system being developed by the Canyon Research Group, Inc. HOPSO is a training system currently being developed at NTEC on the ADCONS which is controlled by a PDP-9 computer.

The details of the development of HOPSO need not be reviewed here since it is the task selection technique which is to be emphasized. In examining the curriculum, we find that it is divided into discrete exercises. Each exercise can consist of a flight task, a lecture, a diagnostic error message, or a specific instruction of some type. Examples of the exercises may be found in Table 1. In scrutinizing Table 1, it can be seen that the exercises shown provide illustrations typical of the Navy's flight training systems.

The HOPSO system possesses two desirable features which make it an adaptive system. First, it attempts to individualize instruction by creating unique trajectories through the curriculum for each student. Secondly, when completed, the system will be able to modify itself (self-organizing) as it gains experience with students.

TABLE 1. EXAMPLE EXERCISES IN THE HOPSO SYSTEM

EXERCISE NUMBER	DESCRIPTION
. . .	
211	STRAIGHT AND LEVEL FLIGHT
221	LEVEL TURNS LEFT
222	LEVEL TURNS RIGHT
231	STANDARD RATE DESCENT
232	STANDARD RATE CLIMB
241	CLIMBING RIGHT TURN
242	DESCENDING LEFT TURN
243	DESCENDING RIGHT TURN
251	LEVEL SPEED ACCELERATION
252	LEVEL SPEED DECELERATION
263	STANDARD RATE DESCENT, SPEED DECELERATION
. . .	

Since all the systems to be discussed are attempting to individualize instruction, it would be well to examine how such a task is accomplished. Figure 5 shows a possible transition matrix for the example tasks which nicely illustrates the requirements of the problem. Suppose that the nth exercise was task number 211: straight and level flight. The student could be required to do task number 221 (a level turn to the left) for the next (n+1) exercise. The transition from task 211 to 221 is represented by the letter A in the matrix. Letter B represents in turn the transition from 221 to 222 and so on. The set A, B, and C represent the student as he moves down the sequence of tasks in the order of their listing. Letter D however, represents a situation where the student jumps from task 231 to 242, and is allowed to skip tasks 232 and 241. If the system allows the student to make unique transitions because of his individual characteristics or history, then this is precisely what we mean by adaptive training. The question then is: on what basis do we present one student with one sequence of tasks and another student a different sequence of tasks?

Individualized trajectories through the curriculum should ideally be based on performance measurements. The HOPSO system measures several performance dimensions. All performance measures are compared with nominal values in such a way as to assign the ratings of good, acceptable, poor, and very poor to four intervals on the performance scales. The data is further reduced to dichotomies (pass-fail) in order to facilitate branching decisions. Most all of the branching logic seems to depend on these single binary dimensions although the more elaborate performance measures could ostensibly be used for diagnostic feedback.

With the performance measures coded into dichotomies, a student would then receive a pass-fail rating for every attempt at each task. It would be desirable to keep a record of the proportion of students passing each task as a function of which task was attempted on the preceding trial. More specifically, the entries in the matrix shown in Figure 6 should contain the percentage of students (having just completed task n) which subsequently complete task n+1. Figure 6 shows a hypothetical set of entries for the matrix wherein the percentages have been converted to probabilities. The entry .81 (listed in the row for task number 212 and the column heading task number 232) represents the idea that 81% of the students having just completed task 212, subsequently complete task 232 successfully. Similarly only 67% of the students that were given task 241 after task 212 were able to successfully complete it. Thus our intuition would tell us that on the occasion that a student has just completed task 212, task 232 would be a better choice for the next task than would task 241. By that same reasoning however, task 222 might be a better choice yet. In the case where a student has just completed task 231, the system might wish to advance him to task 241 and skip task 232. The HOPSO system uses essentially this reasoning for advancing a student through a set of tasks. The process is a bit more complex, however, in the event that a student fails an attempted task. In looking at Figure 5 which records the trajectories of a hypothetical student, it can be seen that the letters above the diagonal represents forward progress of the student and would imply successful completions of the tasks labeled on the rows. For example, the letter E

task n+1

task n	211	221	222	231	232	241	242	243	251	263	271	272
211		A										
221			B									
222				C								
231							D					
232									J			
241												
242											E	
243												
251					I				H			
252												
263												
271									G		F	
272												

$211 \rightarrow 221 \rightarrow 222 \rightarrow 231 \rightarrow 242 \rightarrow \underline{271} \rightarrow \underline{271} \rightarrow \underline{251} \rightarrow \underline{251} \rightarrow 232 \rightarrow 251$

Figure 5. Example Trajectory

task n+1

task n	211	212	222	231	232	241	242	243	. . .
211	1.00	.92	.90	.85	.79	.62	.48	.31	
212		1.00	.93	.88	.81	.67	.52	.40	
222			1.00	.90	.82	.71	.59	.48	
231				1.00	.83	.85	.60	.64	
232					1.00	.86	.90	.60	
.									
.									
.									

Figure 6. Matrix of Transition Percentages

represents the fact that task 242 was just completed successfully and that the student was then moved to the next task with the highest probability of completion (task 271). This transition meant that three intervening tasks were skipped in the process. The letter F represents the event that the student failed task 271 when it was presented which required that it be presented again. Letter G implies that 271 was again failed: an event which sent the student back to task 251. All the entries below the diagonal represent movement backward in the syllabus and thus efforts toward remediation.

The assumptions and learning models behind the branching logic in HOPSO are, of course, not explicit since the logic did not eminate directly from a model. But we may be able to take the logic in parts and make inferences as to what type of assumptions would be necessary. To begin, we will take that part of the logic responsible for the forward progress of the student through the syllabus. In going back to Figure 5, let us assume for the moment that we adopt the strategy of selecting the next task which gives the highest probability of successes. This would mean that we would require the computer to search the row which would correspond to the task just completed by the student, and select the entry (or column) with the highest probability. Let us make a second assumption that all the tasks in the list are monotonically ordered in an ascending order of difficulty. There are several methods by which this might be accomplished (see Airasian & Bart, 1975; Bart & Krus, 1973; Lingoes, 1963). The first three rows shown in Figure 6 illustrate how such a successful ordering might appear. If the order is in fact this successful however, the results of the afore-mentioned branching strategy would seem trivial. In the first three rows of the matrix in Figure 6 it can be seen that in each case the computer would simply select the next task in the sequence and would not affect the skipping of any of the tasks. In this case, we may want to modify the ordering assumption. We will assume that each transition entry t_{ij} (representing the probability of successful completion of task j, given the preceding task i had just been completed successfully) represents an empirical sample estimate of the parameter τ_{ij}. With this we can assume that the monotonic ordering requirement refers to the set of τ_{ij} within a row and not the t_{ij}. The t_{ij} may not be monotonic as in the case of the last two rows shown in the matrix in Figure 6. As can be seen in the row representing transition from task 231, task 232 would in fact be skipped in favor of 241. However, the omission of task 232 would then be the result of sampling error in that the t_{ij} did not follow the same monotonic ordering as the τ_{ij}. Furthermore, if there was no apriori ordering of the tasks (i.e., the designers of the training system did not attempt to order the tasks on an apriori basis and simply let an empirical ordering result) then it would change nothing. This can be seen by imagining the columns represented in random order wherein the branching decisions would be the same.

It would seem as though a branching logic which simply selected the next task to be presented by searching for the one with the highest apparent probability for success, would send the student forward in spurts only to be sent back for remediation. This would be the result of the

student being propelled forward in the sequence at an undue rate due to the random fluctuation of the t_{ij} about the well ordered τ_{ij}. In this case, the rapid rate or progress would not be matched by an appropriate rate in learning and thus periodically the student would be advanced to a point in the sequence in which remediation would be needed. Part of the problem is that advancement in the sequence depends soley on the probability of success on the next item and makes no inferences about the student's learning state. An alternative scheme would be to select the rate of advancement, not on the probability of the next success, but rather on how well the student did on the last task. Presumably, the student's performance on the last task should be related to his state of learning. A good performance should advance him further in the sequence than only a fair performance, though both are rated as a pass or success. A second alternative is to base the selection of the next task on the inferred marginal gain in learning. In other words, the system would estimate how much learning is to be gained from the various alternative tasks, and then select the one with the largest marginal gain. The one with the largest marginal gain may periodically be in a backward direction in the sequence. This would be helpful in designing the backward (below the diagonal) branching logic. This the authors of HOPSO report (Norman, 1977) as being quite complex and still under development. Basing the selection of the next task on the "largest marginal gain in learning" is a principle described in more detail under the alternative techniques.

An additional assumption implicit in the HOPSO system is one of "path independence". By path independence one means that the decision as to the next task is based solely on the previous state or previous task presented. This comes from the fact that the system would restrict its search for the next task to the t_{ij} entries corresponding to the row of the last task completed. Thus the particular path by which the student had come to the just-completed task would be irrelevant. Fast learners would not be differentiated from slow learners. A more expanded explanation of the HOPSO system would show individual differences taken into account in other ways however. Thus the present branching logic would simply look back only one trial to see which task has just been completed, and then look forward to select the next task with the highest probability of success. The path independence assumption is one that HOPSO shares with some other techniques however.

One last assumption of interest in the HOPSO system is that the successful completion of one item is assumed to increase the probability of completion of all the other items. Thus learning one task has positive transfer to other tasks. This also is an assumption which is shared with other techniques. In looking at the column for task 241 in Figure 6 it can be seen that the probability for completion of task 241 increases down the rows representing entries from more advanced tasks. There is no explicit algebraic expression for this positive transfer as there will be in other techniques; and, in fact, the increase is only hypothesized to be there. It seems to be a quite reasonable position, however.

As stated previously, the HOPSO system does not have an explicit learning model from which its selection strategy was derived. But by examining the system's basic configuration, and its implicit assumption, we would be able to say that it implies a certain class of learning models. It is our opinion that in the case of the information type of exercises, and even in the flight maneuvers, the system seems to imply a conceptual or at least discrete-state type of model. This conclusion comes from two prominent features of the branching logic. First, it is the apparent intent of the designers that the student be allowed to skip certain exercises when warranted. Secondly, the probability of completing an exercise correctly is assumed to increase as the student's just-completed exercise is further advanced in the program. Both of these points would lead one to suspect that the implied units-of-acquisition are the underlying skills and cognitive components.

The skipping of certain exercises is desirable in an adaptive system. As it becomes apparent that the subject has acquired a specific skill, you would like to move him ahead to exercises which would tap another skill. Since this is precisely what the HOPSO system is designed to do, it would seem that its intent is to train something other than the units-of-presentation. Additionally, since the focus of the branching logic is on the determination of which exercises should be omitted rather than the determination of the amount of practice or repetition on each exercise, the assumption would be that learning would take place in discrete steps rather than on a continuous basis.

There is one last point that should be noted for later reference concerning the basis of the system's branching logic. The selection of an exercise is based on an attempt to maximize the probability of a success on the next trial. There is no inference of the student's current learning state and thus no attempt to maximize marginal gain in learning.

UNIT-OF-ACQUISITION: THE INDIVIDUAL TASKS

There has been perhaps more development done on this class of models than in the other two categories, and yet it has a lower feasibility for implementation in training situations such as those for which the HOPSO system was designed. These models were designed for situations similar to paired-associate learning. They are characterized by a one-to-one correspondence between the units-of-acquisition and the units-of-presentation.

An example of such a model would be the all-or-none model presented in the introduction. Here the model describes the learning state of a single item relative to a student. It treats each item as an independent unit. Thus in the resultant optimization process, one seeks to maximize the expectation of each item (unit-of-presentation) being in the learned state.

The Random-Trials Increments model represents a second example of this class of models. This model has also been discussed at some length in the introduction and need not be discussed in detail here. It would be good to point out

that like the all-or-none model the RTI model is a description of a specific task or exercise within a specific student. Thus the symbol q_n refers to the probability of a particular student being unsuccessful on the nth attempt of a specific exercise. Learning in this particular case is depicted by a reduction in error probability: thus $q_{n+1} < q_n$.

The RTI model was also designed for paired-associate (PA) type situations (see Norman, 1964). The pertinent characteristics of a PA task are: first, the learning of an individual item comes with repeated presentation; and secondly, the learning of each item is independent of every other item. Empirically, this second point can be debated in that it can be shown that the learning of different items will interfere with each other under certain circumstances. Still, the second characteristic is assumed by the model. Glancing back, the flight training exercises depicted in Table 1 show the resemblance to paired-associate items to be remote at best.

A third model within this category is presented by Atkinson (1976). Like the others, this learning model makes inferences concerning the learning state of a single item or unit-of-presentation. Here, however, the item is not completely independent of the effects of the changing status of other items.

The form of the model is such that there are three learning states. These shall be represented as L, S, and U which denote the learned state, a temporary or short-term state, and an unlearned state respectively. When item i is presented, the following transition matrix T(i) represents possible changes in its state:

$$T(i) = \begin{array}{c c} & \begin{array}{ccc} L & S & U \end{array} \\ \begin{array}{c} L \\ S \\ U \end{array} & \left\| \begin{array}{ccc} 1 & 0 & 0 \\ c_i & (1-c_i) & 0 \\ a_i & b_i & 1-a_i-b_i \end{array} \right\| \end{array}$$

However, between presentations of item i, item j may be presented; in which case i may still change states as summarized in F(i):

$$F(i) = \begin{array}{c c} & \begin{array}{ccc} L & S & U \end{array} \\ \begin{array}{c} L \\ S \\ U \end{array} & \left\| \begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1-f_i & f_i \\ 0 & 0 & 1 \end{array} \right\| \end{array}$$

As one might guess, the matrix T(i) represents the learning process, whereas F(i) represents the forgetting process for item i due to the inference from item j. Basically, when item i is presented, it makes a transition from U to either the permanent state L or the short-term memory state. In making a transition from the short-term memory state, there is presumably a greater chance (i.e., $c_i > a_i$) for making it into the learned state L. The state L is impervious to interference as shown in F(i), by the fact that forgetting only occurs if the item is in the short-term state.

The parameters of the model (a_i, b_i, c_i, and f_i) all carry the subscript i indicating, as was pointed out before, that the model tracks the state transitions of the items rather than the states of the learner. From this Atkinson develops an item-selection strategy which depends dynamically on the parameter estimates for each item within a subject. The results are much like those of the RTI model; i.e., the gain in efficiency increases with successive groups, and the parameter estimates stabilize over items.

There are, of course, other studies in addition to the ones mentioned above which track the ongoing status of the item (see Calfee, 1970; Smallwood, 1971; Karush and Dear, 1966; and Groen & Atkinson, 1966). These models seem to do well in developing reading proficiencies in children. But then it should do well in the training of, say, sight-word recognition wherein the unit-of-acquisition has a one-to-one correspondence with the unit-of-presentation.

UNIT-OF-ACQUISITION: CONCEPTUAL

A discussion of a family of optimization techniques based on models which track the state of the student may be found in a paper by Smallwood (1970). The context is such that we will assume that we are in the process of training an underlying skill or concept; thus, there may be many units-of-presentation to a single unit-of-acquisition. Here, the student is to be given a series of exercises where the exercises may be denoted by the subscripts (1,..., m,..., M). We will further assume that the exercises are ordered along some dimension such as difficulty. Thus, the symbol m would refer to an exercise at the mth level of difficulty. Furthermore, we will assume that in training this particular skill, we can depict the student's progress as transitions along a series of discrete states. This would imply that the unit-of-acquisition is a concept, or a relatively cognitive psychomotor skill. The transitions from state-to-state are assumed to satisfy a Markovian process and can be represented by a transition matrix denoted as T where, in general, the states and their transition probabilities may be labeled as in Eq. (4).

Let us further suppose that we have a set of A instructional alternatives which will change the transition probabilities. It is obvious this would be needed in order to impact the system. Thus, in general, there would be A transition matrices wherein the transition matrix for the ath instructional alternative would be denoted T(a) and the transition probabilities $t_{ij}(a)$.

$$T = \begin{array}{c} \\ S_1 \\ S_2 \\ \vdots \\ S_i \\ \vdots \\ S_j \\ \vdots \\ S_J \end{array} \begin{array}{c} \begin{array}{ccccccccc} S_1 & S_2 & \cdots & S_i & \cdots & S_j & \cdots & S_J \end{array} \\ \left\| \begin{array}{ccccccccc} t_{11} & t_{12} & \cdots & t_{1i} & \cdots & t_{1j} & \cdots & t_{1J} \\ t_{21} & t_{22} & \cdots & t_{2i} & \cdots & t_{2j} & \cdots & t_{2J} \\ t_{i1} & t_{i2} & \cdots & t_{ii} & \cdots & t_{ij} & \cdots & t_{iJ} \\ t_{j1} & t_{j2} & \cdots & t_{ji} & \cdots & t_{jj} & \cdots & t_{jJ} \\ t_{J1} & t_{J2} & \cdots & t_{Ji} & \cdots & t_{Jj} & \cdots & t_{JJ} \end{array} \right\| \end{array} \tag{4}$$

Furthermore, we will assume that there is a discrete set of K responses which may be made while the student occupies the jth state. Thus, we will denote $r_{jk}(a)$ as the probability of the student making the kth response to alternative a while in the jth state. Let the matrix R(a) summarize these probabilities as:

$$R(a) = \begin{array}{c} \\ S_1 \\ S_2 \\ \vdots \\ S_j \\ \vdots \\ S_J \end{array} \begin{array}{c} \begin{array}{ccccc} R_1 & \cdots & R_k & \cdots & R_K \end{array} \\ \left\| \begin{array}{ccccc} r_{11(a)} & \cdots & r_{1k(a)} & \cdots & r_{1K(a)} \\ r_{21(a)} & \cdots & r_{2k(a)} & \cdots & r_{2K(a)} \\ r_{j1(a)} & \cdots & r_{jk(a)} & \cdots & r_{jK(a)} \\ r_{J1(a)} & \cdots & r_{Jk(a)} & \cdots & r_{JK(a)} \end{array} \right\| \end{array}$$

In this general form, the above descriptions could represent a variety of situations. As stated previously, matrix T would best be able to describe the acquisition of a cognitive type skill wherein the acquisition takes place in discrete steps. An example of this might be a skill which could be loosely defined as the ability to hypothetically perform a landing maneuver in simulation. Further, we will suppose that the landing ability is mainly composed of two subprocesses we could simply summarize as stick control and throttle control. Let us also say that some prior work had shown that the instantaneous learning state of the student pilot could reasonably be characterized in a four-state process. Let S_1 depict the terminal state wherein the student has both adequate throttle control and stick control; S_2 depict the state wherein there is adequate stick control

(the student can track the glidepath well enough but comes in too fast or stalls); S_3 depict the case where he attends to the throttle adequately but to the exclusion of the stick; and S_4 be the beginning state wherein the student has neither adequate stick nor throttle control.

The M levels of exercises could be similar to the following:

level 1	exercises in varying speed while holding altitude constant
level 2	exercises in varying altitude while holding speed constant
level 3	landing on a long runway at a slow speed
level 4	landing on a long runway at a high speed
level 5	landing on a short runway at a slow speed
	.
etc.	.
	.

There are, of course, any number of different ways the exercises could vary across levels. The relevant feature is that the exercises differ along various qualitative, perhaps non-quantifiable dimensions which would be linked conceptually to the cognitive states in the learning model. In this example of landing exercises, the exercises differ in ways which would be pertinent to our hypothesis in the model: landing skills are composed of two main subprocesses—stick control and throttle control. Thus our instructional alternatives in this case will be the determination of the next appropriate exercise.

In the present example, we will suppose that the set of K responses are also classified in a qualitative manner such as those listed below:

R_1 Successful landing maneuver (or whatever the exercise calls for).
R_2 Altitude is within tolerance but velocity is not.
R_3 Velocity is within tolerance but altitude is not.
R_4 Both velocity and altitude are out of tolerance.

Of course, the four R_k could also be listed in a quantifiable manner such as: good, fair, marginal, and poor. The response designations listed, however, would be more meaningful in terms of setting up the branching logic, provided it gives an adequate representation of reality.

Figure 7 shows a set of instructional alternatives for the first few levels of exercises. Herein the instructional alternatives are simply branching decisions but the generality of the techniques described allows far more flexibility. For example, the instructional alternatives could refer to the usage of different types of presentation of the same information or usage of different media. Regardless, the instructional alternatives in the present case are utilized as a determination of the branching logic.

In developing a model for illustration, let us assume that nothing is learned about a subprocess (stick control or throttle control) unless that

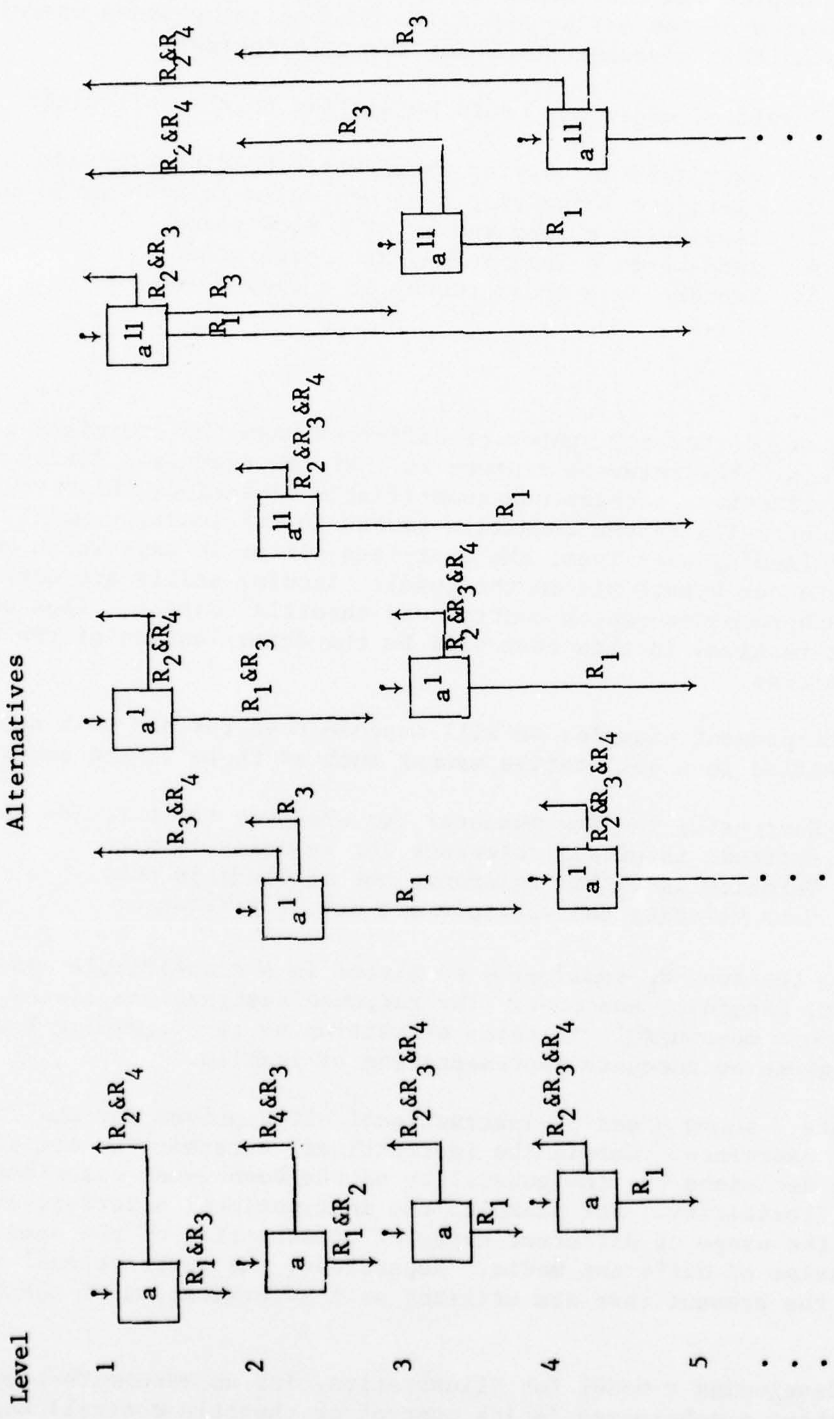

Figure 7. Example Branching Network

subprocess is utilized by the exercise in question. Thus, if a level 1 exercise is utilized (vary speed while holding altitude constant), we will assume that stick control is taxed but throttle control is not, and thus describe the transition probabilities as in

$$
T = \begin{array}{c} \\ S_1(S\&T) \\ S_2(S\&\overline{T}) \\ S_3(\overline{S}\&T) \\ S_4(\overline{S}\&\overline{T}) \end{array}
\begin{array}{c} \begin{array}{cccc} S_1 & S_2 & S_3 & S_4 \end{array} \\ \left\Vert \begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ t_{31} & 0 & 1-t_{31} & 0 \\ 0 & t_{42} & 0 & 1-t_{42} \end{array} \right\Vert , \end{array}
\qquad (5)
$$

where S refers to the stick subprocesses being present and $\overline{S}$ refers to their absence. Similarly, T refers to the presence of adequate throttle skills, and $\overline{T}$ their absence. In looking at the transition matrix in Eq. (5) which would represent the probable effects of a level one exercise, it can be seen that only stick skills are presumed to be learned. Thus, transition from $S_4(\overline{S}\&\overline{T})$ to $S_3(\overline{S}\&T)$ would not take place (i.e., $t_{43} = 0$) because a level 1 exercise ostensibly has little or no chance of teaching anything about throttle control. We could further assume that $t_{31} = t_{42}$, and that this transition probability be given some sort of cognitive referent. We could specify that it represents the joint event that the student realizes he did poorly with respect to stick control, and that he gains insight as to how he should correct for it.

Given that an exercise is given which would tax both subprocesses, the transition matrix T' should be applied:

$$
T' = \begin{array}{c} \\ S_1(S\&T) \\ S_2(S\&\overline{T}) \\ S_3(\overline{S}\&T) \\ S_4(\overline{S}\&\overline{T}) \end{array}
\begin{array}{c} \begin{array}{cccc} S_1 & S_2 & S_3 & S_4 \end{array} \\ \left\Vert \begin{array}{cccc} 1 & 0 & 0 & 0 \\ t_{21} & 1-t_{21} & 0 & 0 \\ t_{31} & 0 & 1-t_{31} & 0 \\ t_{41} & t_{42} & t_{43} & 1-t_{41}-t_{42}-t_{43} \end{array} \right\Vert , \end{array}
\qquad (6)
$$

and additionally the corresponding matrix:

$$
\begin{array}{rl}
 & \begin{array}{cccc} R_1 & R_2 & R_3 & R_4 \end{array} \\
R' = \begin{array}{r} S_1(S\&T) \\ S_2(S\&\overline{T}) \\ S_3(\overline{S}\&T) \\ S_4(\overline{S}\&\overline{T}) \end{array} & \left\| \begin{array}{cccc} 1 & 0 & 0 & 0 \\ r_{21} & 1-r_{21} & 0 & 0 \\ r_{31} & 0 & 1-r_{31} & 0 \\ r_{41} & r_{42} & r_{43} & 1-r_{41}-r_{42}-r_{43} \end{array} \right\| .
\end{array}
\qquad (7)
$$

In looking at the meaning of say r_{21}, it will be recalled that S_2 refers to the stick skills as present but the throttle skills not, while R_1 refers to both velocity and altitude being within tolerance. Thus r_{21} essentially refers to the probability that the student happens to get within tolerance on velocity though the throttle skills are absent.

With the state-to-state transition probabilities specified, we now need to be able to determine the current status of a student at each decision point. Let the vector S_h, where

$$S_h = \| s_1, s_2, \ldots, s_j, \ldots, s_J \|,$$

denote the probabilities of a student occupying each of the J states on trial h. Thus the probability that the student occupies the jth state after the hth trial is s_j. Once our theory is represented by a Markovian process, there are many powerful theorems which allow us the ability to derive the probability of various events. Many of these theorems may be found in Karlin (1966).

Before showing how the elements of the vector S_h might be determined, it would be good to point out that any transition matrix T summarizes the probabilities of transitions from state i to state j over a single step or trial. The elements of T^2 however, summarize the probability of transition from state i to state j over two trials or steps. In general, T^h would be referred to as an h - step transition matrix wherein the elements represent the probabilities of transition from state i to state j in exactly h steps. Further, if S_0 can be referred to as the start vector whose elements s_j refer to the probability of the process beginning in state j, then

$$S_1 = S_0T$$

gives the probabilities of being in state j after the first trial. In general then

$$S_h = S_{h-1}T = S_0T^h$$

which is how S_h would be determined.

In our illustration we should assume the following start vector

$$S_0 = \| \; 0 \quad 0 \quad 0 \quad 1 \; \|$$

since it would be reasonable that our student would begin in the most primitive state. Thus if the level 1 exercise were used first, we would apply the transition matrix T found in Eq. (5) which would result in

$$S_1 = S_0T = \| \; 0, \; t_{42}, \; 0, \; 1-t_{42} \; \|.$$

If the response outcomes were such that our branching logic presented a subsequent exercise which taxed both subprocesses, then T' as found in Eq. (6) would be applied. Thus

$$S_2 = S_1T' = S_0TT' = \| \; t_{42}t_{21}+(1-t_{42})t_{41}, \; t_{42}(1-t_{21}+(-t_{42})t_{42},$$

$$(1-t_{42})t_{43}, \; (1-t_{42})(1-t_{41}-t_{42}-t_{43}) \; \|. \qquad (8)$$

Thus as shown in Eq. (8), we would possibly apply different transition matrices on different trials or steps depending upon which level or type of exercise was given. To further complicate the expression, if a multiple-branch logic such as that shown in Figure 7 were used, then the matrix R' shown in Eq. (7) would also enter in.

Smallwood (1970) points out that in the present context S_h would provide a sufficient history of the student by which the system should make its instructional decisions. This results from our assumption of a Markovian-type learning model in which the probability of current state occupancy is all that is needed. With this, we should consider the student's response probability which shall be denoted $p(R_k|S_h, a)$. This represents the probability that the student makes the kth response given his current status S_h and instructional alternative a. Now, Smallwood includes a in this expression though in our present illustration we would include m instead. This is because Smallwood (1970) worked out the so-called preresponse transition case (see Smallwood, 1967) whereas our illustration would utilize the postresponse transition case. The calculations are similar regardless. We can now derive $P(R_k|S_h, a)$ as

$$P(R_k|S_h,\ a) = \sum_{ij}\sum\{\text{prior state} = i,\ \text{succeeding state} = j,$$
$$\text{kth response, given } S_h \text{ and alternative } a\}$$
$$= \sum_{ij}\sum s_j t_{ij}(a) r_{jk}(a).$$

For our postresponse transition case, we would calculate

$$P(R_k|S_h,\ m) = \sum_j s_j r_{jk}(m)$$

where $r_{jk}(m)$ is the probability of making the kth response while occupying the jth state and responding to the mth level exercise. Smallwood goes on to derive the updated state probabilities s_j' in S_{h+1} as

$$s_j' = \frac{\sum_i s_j t_{ij}(a) r_{jk}(a)}{\sum_{ij} s_j t_{ij}(a) r_{jk}(a)}.$$

The last concept to develop before being able to utilize those functions for optimization purposes is a cost/benefit structure. Let represent the cost of instructional alternative a being presented to a student leaving the mth level. This cost is usually conceptualized as resulting from instructional time. Furthermore we need to assign a cost to the possibilities of the instructional process terminating before being absorbed in state S_1. Let y_j represent the cost of terminating the instruction in state j. Then it would follow that the expected cost at the conclusion of the instructional session would be

$$\sum_j y_j s_j.$$

Now let $W_m(S_h)$ denote the minimum expected total instructional cost for a student with current history (probabilities of current state occupancy) S_h on trial h and having just left the mth instructional level by being presented instructional alternative a. This expression may be formulated in terms of the following recursive equation.

$$W_m(S_h) = \min_a \left\{ \sum_k P(R_k|S_h,\ a) \left[c_{mn} + W_m(S_{h+1}) \right] \right\}. \qquad (9)$$

Additionally define

$$W_m(S_H) = \sum_j y_j s_j$$

as the cost associated with the termination trial H of instruction. As can be seen, $W_m(S_h)$ can only be defined recursively. If $W_m(S_{h+1})$ were known at the time of trial h, then the system could simply calculate the

expected cost of each alternative and select the alternative associated with the minimum cost. As it is, the system must solve for a solution in an iterative fashion. Smallwood (1970) presents an iterative algorithm which seems to be quite efficient in that as a limit, the optimal policy cost function seems to be obtained in just three or four iterations.

The formulation presented by Smallwood is in quite general terms. Some work would still be required before actual implementation. The specific derivations and iterative algorithms would need to be obtained for each model needed. However, this formulation is a beginning for probably the most common of training situations, wherein there are a small number of units-of-acquisition in the form of concepts or cognitive skills. There are numerous models in the literature which would fit into this class, such as the concept-acquisition models (see Trabasso & Bower, 1968; Chumbley, 1972; Wickens & Millward, 1971; Williams, 1971; Millward & Wickens, 1974; Nahinsky, 1970). More will be said later about the specific requirements for implementation.

UNIT-OF-ACQUISITION: CONTINUOUS PSYCHOMOTOR SKILLS

The first technique to be presented in this section is an optimization problem similar to that of Smallwood's, in that it is again the underlying skill that is the unit-of-acquisition and not the exercises themselves. In this technique presented by Wollmer (1976) it is assumed that the exercises are organized into levels. Furthermore, the levels are sequenced according to a strict ordering of difficulty. Let the numbering of the levels be denoted by the symbols (1, ..., m, ..., M) where M is the most difficult level.

For the most general formulation of the model, let there be S_1, ..., S_j, ...S_J learning states which the student may occupy, with the stipulation that S_1 be defined as a terminal state in which the student has just perfomed successfully at level M. The states are ordered in J such that S_J is the most primitive and S_1 the most advanced state. Thus with this model and the formulation to follow, we will only assume forward progress through the levels. The absence of the need for backward movement or remediation could ostensibly be achieved by designing relatively small increments between the levels of instruction. In fact, the technique focuses on determining how many times an exercise should be repeated before going on. Thus, Wollmer goes to some effort to limit this technique to situations wherein there is a true ordering and only advancement in the program is needed.

In the case of the acquisition of textual-type material, one feasible situation would be wherein material covered at one level includes the material covered at preceding levels with the addition of a small amount of additional material. A second situation is one where the material in the different levels is virtually the same, but the amount of prompting is varied. In the case where a specific skill is to be acquired, the exercises at the differing levels may in fact be the same, but the time constraints are tighter at the higher levels.

At this stage, we will assume that the instructional alternatives allowed are simply a selection of the next exercise level for presentation. If that exercise is failed, the system will simply return the student back to that same exercise for repetition.

To formulate the response probabilities, let the matrix

$$
L = \begin{array}{c} \\ S_1 \\ S_j \\ S_J \end{array}
\begin{array}{c} 1, \ldots, m, \ldots, M \\ \left\| \begin{array}{ccccc} p_{11} & \cdots & p_{1m} & \cdots & p_{1M} \\ p_{j1} & \cdots & p_{jm} & \cdots & p_{jM} \\ p_{J1} & \cdots & p_{Jm} & \cdots & p_{JM} \end{array} \right\| \end{array}
$$

where p_{jm} is the probability that the student will perform correctly on an exercise at the mth level when he is currently in the jth learning state. Thus if the student is in state j and we wish to present exercise M (the most difficult one), then we have two possible outcomes of this action. The student will either perform adequately on exercise M with probability p_{jM} or he will fail with probability $1-p_{jM}$. If the performance is in fact correct, then the student is said to move to learning state S_1. The following transition matrix summarizes possible transition outcomes for the presentation of exercise M:

$$
\begin{array}{c} \\ S_1 \\ S_j \end{array}
\begin{array}{c} \begin{array}{cc} S_1 & S_j \end{array} \\ \left\| \begin{array}{cc} 1 & 0 \\ p_{jM} & 1-p_{jM} \end{array} \right\| \end{array}
$$

In general, if we had presented exercise $m < M$ we would have

$$
\begin{array}{c} \\ S_j+1 \\ S_j \end{array}
\begin{array}{c} \begin{array}{cc} S_{j+1} & S_j \end{array} \\ \left\| \begin{array}{cc} 1 & 0 \\ p_{jm} & p_{jm} \end{array} \right\| \end{array}
$$

Thus far, it looks as though the best choice of exercises for a student in state S_j would be to select the exercise corresponding to the maximum

$$\max_m p_{jm}$$

listed. This could be found by simply searching through the row for S_j in matrix L. This would be similar to the algorithm used by the HOPSO system with the exception that in the HOPSO system, the row would be in reference to the previous task completed rather than the current learning state.

An additional difference would lie in the fact that there is a functional relationship between the probabilities of being successful on an exercise while occupying different states. Let $p_{j+1,m+1}$ be the current probability of being successful on exercise m+1 while in S_{j+1}. But the student is presently in S_j whereupon he successfully completes exercise m and moves to state S_{j+1}. Now the probability of getting the next exercise (m+1) is

$$\begin{aligned} p_{j+1,m+1} &= p_{j,m+1} + q_m(1-p_{j,m+1}) \\ &= (1-q_m)p_{j,m+1} + q_m \end{aligned} \tag{10}$$

where q is defined as the probability of being able to perform exercise m+1, given that he could not do m+1 before, but has just completed exercise m. The scalar q_m acts as a parameter depicting the amount of transfer between exercise levels. Note that the probability of the student being able to do exercise m+1 is greater when he is in an advanced learning state; i.e., $p_{j+1,m+1} \geq p_{j,m+1}$.

Say that we denote the cost of the presentation of exercise m as c_m. Let π refer to an instructional policy, an adaptive logic dealing more with the repetition of exercises than branching from one to the other. Then let $V(\pi,s_j)$ be the total expected cost under logic π when the student is in state s_j. Then we could define $V(\pi,s_j)$ recursively as

$$V(\pi,s_j) = c_m + p_{jm}V(\pi,s_{j+1}) + (1-p_{jm})V(\pi,s_j). \tag{11}$$

The problem then reduces to deducing a logic π such that $V(\pi,S_j) < V(\overline{\pi},s_j)$ for all s_j and all $\overline{\pi}$.

As shown in Eq. (10), if the student performs exercise m correctly while in state j, then the probability of performing m+1 becomes $p_{j,m+1}+q_m(1-p_{j,m+1})$ where the expected number of trials required to complete m correctly was $1/p_{j,m}$ at a cost of $c_m/p_{j,m}$. This would mean that if we require k correct responses to m before going to exercise m+1, then $p_{j,m+1}$ becomes

$$p_{j,m+1} = 1 - (1-q_m)^k (1-p_{j,m+1}).$$

Thus if logic π requires $k(m)$ repetitions of m then Wollmer obtains

$$V(\pi, s_j) = \sum_m \frac{k(m)c_m}{p_{j+1,m}}$$

From here, Wollmer proceeds to search for

$$\min_\pi V(\pi, s_j).$$

In this search, Wollmer finds some interesting solution properties; but as one might guess, he cannot obtain an analytic solution to the problem. Thus Wollmer presents a general algorithm wherein one could solve dynamically for $k(m)$ (the number of repetitions required at each level) which would minimize instructional cost.

Wollmer's model provides an interesting complement to the Smallwood (1970) solution. Wollmer assumes that the nature of the skills to be trained must be drilled into the student by repetition. Smallwood's formulation would better be able to deal with a type of branching logic wherein some exercises may be presented only once or not at all. Smallwood's branching logic would of course be best when the underlying skills are conceptual or cognitive in nature and the branching can be based on inferences about the concepts acquired. Chant & Atkinson (1973) developed a set of techniques which in many ways is similar to Wollmer's. Their techniques were to determine what they called an optimal allocation of instructional effort to interrelated learning strands. By learning strands they had meant blocks of material which could be categorized into meaningful units. The present discussion will present a variant of this application which it is felt will have more utility in the training of psychomotor skills. This technique does not assume that there is a well-documented cognitive model or explanation of the process. It assumes merely that a reasonable description of the learning curve exists and that there is an assessment procedure by which the current point on the learning curve can be determined.

To begin, we will assume that it has been determined that a certain segment of flight training consists of M separate but related skills. Furthermore, let it be assumed that these skills are acquired by practice. Thus the training is relatively simple: one must allocate a certain amount of the total training time to each of the M skills or tasks which must be practiced. Let us again assume that the M skills are ordered in difficulty such that skill or task M is the most difficult and thus would be presented last.

In taking any two skills, say m and m+1, they are assumed to be interdependent to the point that, the more practice the student has on task m pertaining to skill m, the more positive transfer will result when the student attempts the next task m+1. This is the case where it could be said that the M tasks represent a set of non-orthogonal skills where the number of basic underlying independent factors are less than the number of skills being trained.

Let x_m represent the achievement level of a student on task m after a certain amount of training. Additionally let $f(x_m, y)$ denote the instantaneous learning rate on task m. The arguments of $f(x_m, y)$ indicate that the instantaneous learning rates are a function of the achievement level for task m and a composite (y) of the effects of the current achievement levels of other tasks. For the present illustration we will assume that y refers simply to the interdependence of task m with the next task which is m+1. With this simplification, the interdependency between tasks m and m+1 can be shown in Figure 8. It is clear in the figure that the instantaneous learning rates of both tasks are dependent on the difference between their achievement levels i.e., $y = (x_m - x_{m+1})$. Thus, as training task m begins to push x_m ahead of x_{m+1}, the learning rate $f(x_m, y)$ diminishes while the positive transfer causes $f(s_{m+1}, y)$ to increase.

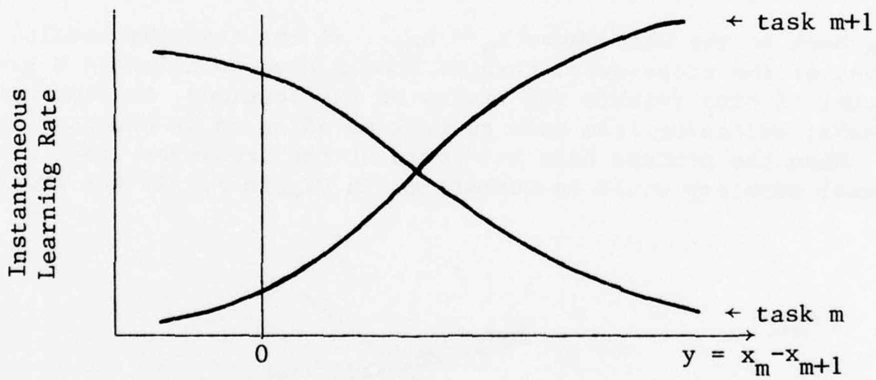

Figure 8. Learning Rate Characteristics

The problem in this situation is to determine when to terminate practice on task m in order to begin practice on task m+1; when to shift back to m, back again to m+1, and so on; so that over a fixed period of time you attain a maximum weighted composite achievement level for the two skills within a given amount of time. In other words, the total practice time within a block is assumed fixed but the amount of time allocated to each task is variable.

An approximation to the solution may be affirmed by examining Figure 8 and simply using intuition. Say that $y = x_m - x_{m+1} = 0$ wherein the student's achievement level on both tasks is identified. As can be seen in the figure, the learning rate for task m is higher at this point (it may be

remembered that m is the easier of the two tasks in our rank ordering) but beginning to decline. Thus having the student practice task m would seem to be more profitable. But as the student practices task m, the achievement level x_m will advance relative to x_{m+1} making $y = x_m - x_{m+1}$ positive. If we pick a point on the abscissa wherein y is to the right of the cross-over, then it becomes advantageous to have the student practice task m+1. In practicing task m+1, the difference $y = x_m - x_{m+1}$ again diminishes. Only at the point of the cross-over does it seem to be equally advantageous to present both tasks. If both tasks are of equal benefit to us, then Chant & Atkinson show (in much more detail) that our conclusion from above is essentially correct. To generalize this a bit more however, let the relative benefits of the two skills be represented by the constants b_m and b_{m+1}. Then the objective expression to be maximized is

$$b_m x_m(T) + b_{m+1} x_{m+1}(T), \tag{12}$$

where $x_m(T)$ represents the final achievement level of the mth skill. Thus the cross-over point would represent the solution wherein $b_m = b_{m+1}$ but $b_m > b_{m+1}$ would hint that the point wherein training on m would be equally advantageous as m+1 might be to the left of the cross-over, and vice versa for $b_m < b_{m+1}$.

Going back to the case where $b_m = b_{m+1}$, if the training session began at the point of the cross-over, then we should have to allocate a proportional amount of time (within the limits of the session) to instruction on both tasks, switching from task to task as often as is practically possible. When the process does not begin at the crossover, then our instructional strategy would be summarized in Figure 9. If the process

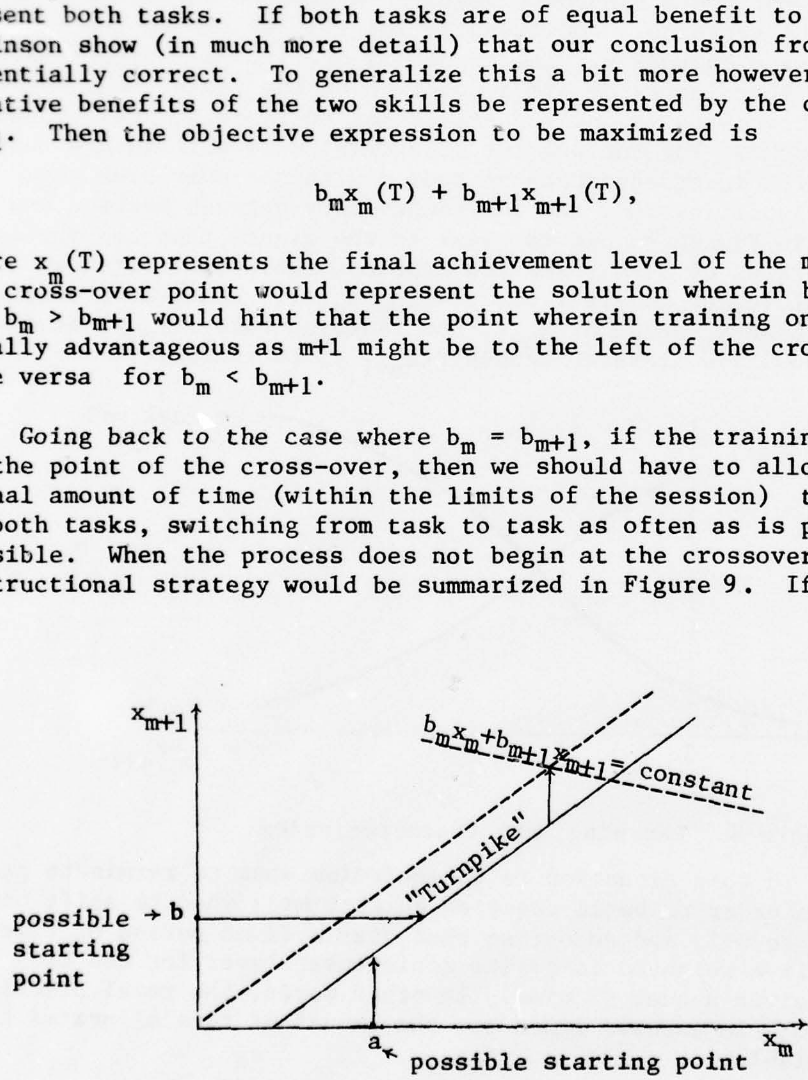

Figure 9. Turnpike in an x_m, x_{m+1} plane. Optimal trajectories shown for two possible starting points.

begins at point a, then the optimal solution would have us to only present instruction on skill m+1, until reaching what is referred to as the "turnpike". The turnpike represents the solution wherein both skills m+1 and m are given instructional time or effort according to some ratio. Thus as indicated by the diagonal trajectory, both m+1 and m are advancing in achievement. Further along the turnpike, the optimal strategy is to leave and again give training only on m+1 until a composite criterion (as in Eq. (12) in achievement is met. At this point, the optimal solution may not be exactly what would be expected from our simplistic intuitive discussion, but it is close.

Figure 9 represents a solution to a deterministic model, although Chant & Atkinson go on to show that the stochastic version has the same general solution. The authors give this solution for only two skills or, in their case, learning strands. They point out however that the extension to three or more skills should be relatively straightforward. Hence in its most general form, the Chant & Atkinson solution could be adapted to skill training where we can assume a continuous learning curve. With performance measures (on which we will have more to say later) designed to estimate learning rates at strategic points in training, the technique offers a way to maximize a weighted composite of achievement levels within a fixed amount of instructional time.

SECTION IV

FEASIBILITY

In comparing the various techniques for possible implementation, several pertinent questions must be considered. One must ask whether the assumed learning models are suitable for the training situation. One also needs to examine the specific function which is to be optimized, and with respect to what. There are problems with respect to parameter estimation and performance measurements with which we must deal. And lastly, just how close is the technique to actual implementation, and what remains yet to be done? This section briefly compares the techniques and examines these points.

MODELS ASSUMED

All the techniques assumed a learning model, even if its description was in the most general terms. Only the HOPSO system did not explicitly assume a model. The HOPSO system seems to imply the type of model in which the units-of-acquisition are the underlying cognitive skills. They are surmized to be cognitive in that it is emphasized that a student's trajectory through the syllabus may involve considerable skipping around. In contrast, the training of non-cognitive motor skills usually emphasizes learning by repetition. A paradox arises however, in that the branching logic seems to work only at the task level and does not attempt to make any inferences about the state of the underlying units-of-acquisition. In other words, it seeks only to maximize performance on the units-of-presentation instead of inferred learning states.

The other techniques can first be organized as to whether the models make state-occupancy inferences about items within a subject, or the subject's state occupany relative to a specific underlying skill or concept. Norman's RTI model and Atkinson's three state model with a short-term memory state, represent models of the first type. These models assume the items, or units-of-presentation, to be important in their own right and thus seek to maximize the probability of their being in the most advanced state by the end of training. This formulation is probably not adequate for reinterpretation in terms of flight training skills. There is most likely positive transfer between the exercises involved in flight training in contrast to the item (exercise) independence assumed by the RTI model, and the interference assumed in the forgetting model. Still the techniques based on these models should be kept in mind in the event that something like training in language or terminology skills (for which the techniques were originally designed) is required.

Of the last three techniques, the one by Smallwood (1970) seems most applicable to cognitive skills. In its general form, it simply assumes that a student can occupy one of several discrete states relative to an underlying skill or unit-of-acquisition. This generally opens up the

possibility for various models already proposed. He proposes that these states are ordered, but a strict ordering is not necessary. These states could represent qualitative stages in conceptual development which would be rich in information needed for the development of a branching logic. It is our view that this generic form of a model represents the type which could depict the kind of curriculum with which the HOPSO system deals.

The last two models represent more of a non-cognitive type of skill training. Wollmer's model assumes a strict ordering of the exercises and states. Wollmer gives conceptual examples but focuses more on repetition of exercises rather than the type of branching Smallwood's technique would give. Chant & Atkinson's technique would be similar: they propose a possible application with cognitive material and yet the model simply assumes that the subject occupies a point on a learning curve. The learning curve, of course, could depict most any type of learning, but represents much more of an approximation than the more specific models. Still, learning curve assumptions are very applicable for some pure motor skills, and may even be a good point of departure for the more cognitive skills when one lacks a specific model.

OPTIMIZATION

It will be recalled, that the basic form of any optimization procedure is to explicitly state just what it is that is to be maximized (or minimized) with respect to some kind of instructional alternatives. Thus two points should be compared on the different techniques: first, the expression to be optimized; and secondly, the instructional alternatives which allow the optimization to take place.

For the HOPSO system, the RTI model, and Atkinson's short-term model, the instructional altervatives by which functions are optimized are simply the item or task selections themselves. The HOPSO system seems to base its task selection on a maximization of the performance on the next item presented. The RTI model selects the item which would result in the maximum gain in learning (reduction in error probability q_n). Atkinson's short-term memory model attempts to select the item i which would yield the largest gain in the probability of transition to the learned state.

Smallwood's technique considers more than simply item selection. In general terms, he refers to an optimization with respect to unspecified instructional alternatives. In the simplest sense, instructional alternatives could be just item selection: but, the generality of his formulations suggest that they may be mini-branching schemes in their own right, as suggested by Figure 7 . Further, choices in the alternatives a are sought which would obtain

$$\min_{a} W_m(S_h)$$

where $W_m(S_h)$ is defined in Eq. (9). In words, Smallwood seeks to minimize the cost by considering the alternatives at the point of trial h, given a student whose history is characterized by (S_h). It is cost that is minimized rather than learning that is maximized, the assumption is that some of the instructional alternatives would be more costly than others. Learning is taken care of by assigning a cost to the event that the student terminates while in an intermediate state.

Wollmer's technique seeks a policy π which would

$$\min_{\pi} V(\pi, s_j)$$

as defined in Eq. (11). The instructional policy (π) here refers to the number of repetitions of the exercises at each level m. In words, he seeks to minimize the cost (or time) required to get the student to the terminal state by an appropriate choice of a policy π on repetitions. The Chant & Atkinson technique, on the other hand, would seek to maximize learning on different exercises (or strands) in terms of maximizing a weighted composite of achievement levels. This is done by choices concerning differential allocation of instruction on the exercises.

PERFORMANCE MEASUREMENT AND PARAMETER ESTIMATION

In discussing performance measurement, we will limit the discussion to only those measurements directly needed in the adaptive logic of the system. Thus at this point, ancillary measures such as student attitudes, or diagnostic information which is fed back to the student (but does not enter into the item-selection algorithm) is not considered. We will consider the measures needed by the system which will develop an adequate summary of the student for adaptive decisions. In most cases, this reduces to a problem in parameter estimation.

In most applied settings, a basic measurement problem is the existence of a small population from which one must sample. Often times, the number of parameter estimates needed outnumber the number of sample observations. Thus, a basic goal in dealing with models is to keep the number of parameters estimated at a minimum, and to pool data whenever possible in order to gain stable estimates.

In the general description of the HOPSO system, it will be recalled that a matrix of transition probabilities, such as in Figure 6, needed to be estimated. If carried to its logical extreme, there would be a transition probability for every exercise paired with every other exercise. Thus if there were M exercises, one would need M^2 estimates (t_{ij}) of the τ_{ij}. If only the forward part of the matrix were estimated, one would need M(M-1)/2 estimates. The HOPSO system plans to force the students through the curriculum until enough of the transition probabilities are estimated so that an empirical determination of the adaptive logic can be determined. The problem here is that with the trial-by-trial information divided so

thinly over the large number of transition combinations, it will take a considerable number of students before the estimates stabilize. A second major problem is that there is no means by which an empirical estimate of a particular transition (t_{ij}) can be made prior to the time that the first student actually attempts it.

Atkinson and Paulson (1972) report an innovative parameter estimation technique which is based on the work of Rasch (1966). They were faced with the problem that in the RTI model, there are at least two parameters to estimate: a and c. But not all the items are of the same difficulty level and not all the students are of the same aptitude. Thus with N items and I students, there exists the difficult task of estimating N·I estimates of a and N·I estimates of c. In estimating the parameters, they essentially used an analysis of variance technique wherein one estimates N subject effects and I item effects with no interaction assumed.

In developing this on parameter c alone, let an analysis of variance model

$$E(c_{ij}) = m + d_i + b_j$$

depict a fixed-effects subject-by-items analysis. Here m is a constant, d_i is the difficulty of item i and b_j is the aptitude of student j. Now c_{ij}, being a probability, needs to be bounded by 0 and 1 for which there is no guarantee in the model. Thus, the parameter c_{ij} was changed to an odds ratio of the form $c_{ij}/(1-c_{ij})$ with the assumptions that: first, the odds ratio is proportional to student ability b_j; and secondly, the odds ratio is proportional to item difficulty d_i. This is expressed as

$$\frac{c_{ij}}{1-c_{ij}} = k\frac{b_j}{d_i}$$

where k is the constant of proportionally. Now taking the log of both sides yields

$$\log \frac{c_{ij}}{1-c_{ij}} = \log k + \log b_j - \log d_i$$

wherein the log of the odds ratio is referred to as the logit. Now Eq. (13) begins to look like an additive model of the form

$$\text{logit } c_{ij} = \mu + \alpha_j + \beta_i,$$

where $\mu = \log k$, $\alpha j = \log b_j$, and $\beta = - \log d_i$. Now the model requires only N+I parameters to be estimated rather than the N·I parameters as before. This means that the extra observations from the subject-by-item

combinations can be pooled to form more stable estimates of the N+I parameters. More importantly, estimates of a student's performance on items he has never encountered can now be estimated.

Fischer (1973) generalized Rasch's work on the Linear Logistic Test Model to the extent that it would seem possible to apply it to some other models as well. Fischer shows that a main effect in the additive model (such as β_i) may be interpreted as the effect of factor i where factor i may represent an underlying skill factor. This would fit in well with our notion that the units-of-acquisition, which are fewer in number than the units-of-presentation, may be considered as underlying skills.

Both the RTI model and Atkinson's short-term memory model have made use of the above techniques. For the RTI model, 2(N+I) parameters need to be estimated (considering both c_{ij} and d_{ij}), while 4(N+I) parameters need to be estimated for the short-term memory model (considering a_{ij}, b_{ij}, c_{ij}, and f_{ij}). In both cases, empirical results showed the estimates stabilizing quickly, and the system gaining in efficiency as it obtains experience with an increasing number of students. Presumably it would be worthwhile to investigate the possibilities of using these techniques with the other models as well. The Smallwood and Chant & Atkinson papers do not show expressions for their parameter estimates, but Wollmer derives a series of maximum likelihood estimates. Wollmer however assumes, as presumably do the Smallwood and Chant & Atkinson papers, homogeneity of parameters across students.

One last point regarding performance measurement should now be discussed. Regarding the adaptive logic within the optimization techniques presented, performance measurement simply reduces to parameter estimation. In raw form, this usually takes the form of error-success data. Even the HOPSO system simply uses error-success data to make branching decisions. Those techniques based on learning models however, derive specific transformations from the error-success data to estimate the parameters. It will be recalled from the introduction, that the incremental model simply used a count of the number of times an item had been presented; whereas the all-or-none model kept a count of the number of successes since the last error. Hence, the learning models serve a useful function in determining the type of performance measures needed by the branching logic. This is not to preclude however, the use of other performance measures for other types of decisions.

IMPLEMENTATION

Of all the techniques listed, the HOPSO system is probably the closest to actual implementation as development is taking place at this time. Use of the RTI model and Atkinson's short-term memory would also take little effort toward implementation. Both techniques were developed from specific models and have previously been used in adaptive training systems. The technique presented by Chant & Atkinson (1973) in the form of optimal allocation of instructional effort to interrelated learning strands has

also been used in an adaptive training system. The paper by Chant & Atkinson did not develop their techniques beyond two strands, but they contend that an extension to three or more would be relatively straightforward. There may be some developmental work needed however, in terms of adapting the technique to interrelated skills as is proposed.

Wollmer's technique for the training of non-cognitive skills, as well as Smallwood's technique for more complex branching, will require additional development before implementation can take place. Both of these techniques were formulated in the most general form as can be seen by the fact that the functions were not specified. The advantage in this is that these techniques are of quite general applicability, but their generality precludes immediate implementation.

One last point concerning the requirements for implementation concerns the cost/benefit structure. Most training systems have ignored this feature and have assumed by default that all information or exercises presented are of equal importance or benefit. In terms of cost, most systems simply attempt to minimize the student's time in training. The optimization techniques offer much more powerful options however. If we can specify a quantifiable utility function regarding the skills trained, and if we can specify a function representing instructional cost, then we actually have much more opportunity for cost-effectiveness in our adaptive logic. If we can't specify cost/benefit functions, then they can always default to the previous assumptions of equal benefit and that cost is defined in instructional time, and be no worse off than before.

SECTION V

CONCLUSIONS

The design of an adaptive logic is really the heart of an automated adaptive training system. If automated training in general is cost-effective, then it is so because of what savings automation brings to the instructional environment. But if automated adaptive training is more cost-effective than its non-adaptive counterpart, it is so because of what savings the adaptive logic brings to the system. The design of the exercises, the performance measurement, and ultimately the measurement of costs and benefits, all owe their utility to an effective adaptive logic. Thus it almost goes without saying, that an investment in the development of an efficient adaptive logic in a training system is worth considerable effort. The main question then is: how much developmental effort would be required to make use of some of the techniques developed in this report on both the short-term and long-run? Also, what would be the recommended points of departure for these efforts?

SHORT-TERM DEVELOPMENTS

At this point in time, automated adaptive systems are presently under development. Systems such as HOPSO are presently developing branching logics for complex curriculum. Thus the immediate need is suggestions for even minor modifications which will allow optimization of some of the components within the system.

After examining quite a few optimization techniques, it is our conclusion that Smallwood's (1970) general formulation offers the best point of departure for the flight training systems being developed. It will perhaps take further development, whereas some of the techniques are closer to implementation, but it would be much more general in the variety of situations it could handle. In simple terms, it allows us to be able to postulate that the student may occupy several unspecified states. It further allows us to make selections from among a set of undefined instructional alternatives at a variety of levels. In this general form, implementation in a variety of training systems would be possible without a lot of development on each training system.

As an illustration, any training syllabus is first designed on an a priori basis. That is, the first edition of the syllabus is non-empirical. If the syllabus contains a fixed branching logic, we might well want to ask the designers or instructional team for the basis on which their instructional decisions were made. When asked about a particular branching decision, instructors say something like, "If the student responds like this—then I suspect that he doesn't quite have skill (or construct) x down, even though he may have skill y mastered. Thus it is surmized that exercise A would be profitable." From the above statement, the instructor is implicitly making inferences about the

student's state of learning (e.g., "he doesn't quite have skill x but does have y") and then proceeds to make decisions by which he feels the student could make maximal gains (e.g., "exercise A would be profitable").

The developers of an initial training system could sit down with an instructor, divide the curriculum into relatively small blocks, and then ask the instructor what states (within a block of exercises) would the student be likely to occupy. Furthermore the instructor could be asked what alternative branching schemes within the block of exercises would be plausible. This the system designers would use to construct an initial mini-model within each block of exercises.

Once a mini-model is constructed and maybe revised slightly by the system designers for technical reasons, Smallwood's optimization technique could begin to take over: obtaining estimates of parameters, and empirically determining which of the instructor's recommended alternatives should actually be selected so that the student's advancement in the states is optimized. The question might now be asked: what if the instructor's model is invalid and the original branching alternatives he gave us do not reflect the best possible? Then we would have to answer that we should be no worse off than we were before using the automated system, because we were only using the framework which the instructor used previously. What is needed of course, is a long-range research program which would develop valid learning models from which to begin. The point here is that Smallwood's technique could be general enough to use the instructor as a point of departure until the long-range objectives are fulfilled.

Developmental Steps	Time Required	Levels of Effort in Man/Months
Derivation of Specific Solutions	Two Months	One
Initial Software Development	Four Months	Three
Analytic Evaluation and Adjustments	Three Months	Two
Short Empirical Evaluation	Three Months	Two

Figure 10. Developmental Steps and Levels of Effort in the First Stage of Development of a Demonstration Package

In order to ultimately implement the techniques advocated in this report, it would be suggested that a two-stage development program should ensue. The first stage would entail the development and analytic evaluation of the algorithms and resulting software. The second stage would

entail incorporation of the software from the first stage, into an on-going training system for evaluation with actual student pilots. Figure 10 shows the steps required in the first stage and the associated levels of effort. As can be seen, there would be four basic steps in the first stage of development. The first step would be essentially mathematical. Here Smallwood's basic solutions would be extended to models of various representative forms. The second step would involve the translation of the mathematical solutions into operational software. In the third step, the resulting algorithms would be evaluated analytically via Monte Carlo runs. Data from simulated stat-students could be used for evaluation of specific problem areas in the solutions. Specifically the iterative solutions should be evaluated as to rate of convergence on parameter estimates as well as convergence to the optimal policy regions. Additionally, the solutions need to be evaluated as to their robustness concerning their assumed learning models. This can easily be evaluated by varying the properties of the models used for the data generation of the stat-students. In the final step, simple training tasks for actual students could be devised to evaluate certain problem areas for the algorithms. The end result of the evaluations would be a revised demonstration package which would then be incorporated in the second stage into an operational flight training system for evaluation with student pilots.

LONG-RANGE DEVELOPMENTS

The ultimate objective would be to fully understand the dynamics of learning, such that truly optimal techniques could be deployed. Toward that goal, development of a generic class of models, which would adequately represent the process of such things as flight training, should ensue. If the short-range implementation of mini-theories were implemented, the dynamic characteristics of the optimization techniques would begin to collect data on the adequacies of the models. This of course would be quite efficient as a developmental technique. Additionally, a good portion of the basic/cognitive research becomes relevant as the evaluation of general forms of models take place.

To the extent that state-occupancy in a model is actually observable, less pressure is put on iterative procedures for parameter estimation. As the states are continually redefined, transitions become more observable and diagnostic branching becomes more exact. The observability of states is precisely what is needed in the acquisition of highly cognitive skills. Here the underlying skills and concepts are linked by a hierarchy or complex network. To the extent that a model can predict state-occupancy or to the extent that the states are observable, the system can optimize quite effectively.

Further work needs to be done on a generic class of models which could modify themselves. An example of this is seen in the RTI model. Here, it will be recalled, the RTI model represented a cross between an

incremental model and an all-or-none model. The RTI model, in the extremes, became all-or-none or incremental when the parameters took on certain values. Since the parameters were determined empirically in a student-by-item design, the RTI model would itself adapt to the uniqueness of the student-item combination. Thus if a particular item or task were primiarily conceptual the RTI model would approach an all-or-none solution. But if the next task were acquired in an incremental fashion, the parameters would adjust and the model swings back in the other direction. Thus the specific form of the model was determined by the data. What is needed is for this self-modification property to be developed in some of the future models. The generality of some of the models presented shows that self-modification is present to a limited extent already. This properly needs to be extended however for one very good reason. Basic research in the applied training environments, such as flight training, is quite expensive. The student populations are small and have little spare time to take part in experiments. Additionally, the training equipment utilization is difficult to get for basic research purposes. Since the basic research is needed, it would seem to be cost-effective to have the training systems themselves collect the needed data. By formulating the models in terms of a general structure, the incoming data would modify the models, resulting in an increase of efficiency, while collecting information concerning basic learning axioms in the applied setting. Again this can only be done to the extent that the states are quite observable.

A final long-range objective, is to determine techniques by which cost/benefit structures can be obtained. The optimization techniques presented offer the possibilities of a large savings in student training time, if it is time that is being minimized in the functions. The techniques offer much more impressive results than this however, if it is a cost function that is minimized. Thus at this point in time the optimization techniques offer more than we are presently in a position to take advantage of. If cost/benefit structures can be eventually obtained, and general self-modifying models designed, then dramatic savings in the field of automated training could be realized over the long-term.

When one looks at the reasons for automation in general, it has two major advantages. The first advantage is the savings obtained as the instructor is to some extent replaced. The second comes from the instructional efficiency gained from the tremendous computing power of the system. That computing power is not utilized unless it can dynamically adapt to the student-task situation. Further, gains in efficiency over the long run are only going to come as our general understanding of the learning process (model) is increased, and as the design of the adaptive logic is capable of utilizing this advancement in knowledge.

REFERENCES

Airasian, P.W., Bart, W.M. Valedating a priori instructional hierarchies, Journal of Educational Measurement, 1975, 163-173.

Atkinson, R.C. Adaptive Instructional Systems: Some Attempts to Optimize the Learning Process. In D. Klahr (Ed.), Cognition and Instruction. New York: Halstead Press, 1976, 81-108.

Atkinson, R.C. Computerized instruction and the learning process. American Psychologist, 1968, 23, 225-239.

Atkinson, R.C. Ingredients for a theory of instruction. American Psychologist, 1972, 27, 921-931.

Atkinson, R.C., and Paulson, J.A. An approach to the psychology of instruction. Psychological Bulletin, 1972, 78, 49-61.

Bart, W.M. Krus, D.V. An ordering theoretic method to determine hierarchies among items. Educational and Psychological Measurement, 1973, 33, 291-300.

Bush, R.R., and Sternberg, S.H. A single operator model. In R.R. Bush and W.K. Estes (Eds.), Studies in Mathematical Learning Theory. Stanford: Stanford University Press, 1959.

Calfee, R.C. Role of mathematical models in optimizing instruction, Scientia, 1970, 105, (697-698), 317.

Chant, V.G., and Atkinson, R.C. Optimal allocation of instructional effort to interrelated learning strands. Journal of Mathematical Psychology, 1973, 10, 1-25.

Chant, V.G., and Luenberger, D.G. A mathematical theory of instruction: instructor/learner interaction and instruction pacing. Journal of Mathematical Psychology, 1974, 11, 2, 132-158.

Chumbley, J.A. A duoprocess theory of concept learning. Journal of Mathematical Psychology, 1972, 9, 17-35.

Conway, E.J., and Norman, D.A. Adaptive Training: New Directions. Proceeding of the 7th NTEC/Industry Conference. 19-21 November 1974. NAVTRAEQUIPCEN IH-240, 145-155.

Crothers, E.J. Learning model solution to a problem in constrained optimization. Journal of Mathematical Psychology, 1965, 2, 19-25.

Dear, R.E., and Atkinson, R.C. Optimal Allocation of Items in a Simple, Two-Concept Automated Teaching Model. In John E. Coulson (Ed.), Programmed Learning and Computer-Based Instruction. New York: Wiley, 1962.

Dear, R.E., Silberman, H.F., Estavan, D.P., and Atkinson, R.C. An optimal strategy for the presentation of paired associate items. Behavioral Science, 1967, 12, 1-13.

Eddowes, E.E. A Cognitive Model of What is Learned During Flying Training. Technical Report FAHRL-TR-74-63, Brooks Air Force Base, Texas, 1974.

Fischer, G.H. Linear logistic test models as an instrument in educational research. Acta Psychologica, 1973, 37, 359.

Fletcher, J.D. Models of the Learner in Computer Assisted Instruction. Technical Report NPRDC-TR-76-23, Navy Personnel Research and Development Center, San Diego, California 92152, December, 1975.

Groen, G.J., and Atkinson, R.C. Models for optimizing the learning process. Psychological Bulletin, 1966, 66, 309-320.

Howard, R.A. Dynamic Programming of Markov Process. New York: Wiley, 1960.

Karlin, S. A First Course in Stochastic Processes, New York: Academic Press, 1966.

Karush, W., and Dear, R.E. Optimal stimulus presentation strategy for a stimulus sampling model of learning. Journal of Mathematical Psychology, 1966, 3, 19-47.

Kelley, C.R. What is adaptive training? Human Factors, 1969, 11(6), 547-556.

Lingoes, J. Multiple scalogram analysis: a set theoretical model for analyzing dichotomous items. Educational and Psychological Measurement, 1963, 23, 501-524.

Lorton, P. Computer-Based Instruction in Spelling: An Investigation of Optimal Strategies for Presenting Instructional Material. Unpublished doctoral dessertation Stanford University, 1972.

Millard, R.B., and Wickens, T.O. Concept-Identification Model. In D. Krantz, R. Luce, R. Atkinson, and P. Suppes (Eds.), Contemporary Developments in Mathematical Psychology, Volume 1, San Francisco: W.H. Freeman, 1974.

Nahinsky, I.D. A hypothesis sampling model for conjunctive concept identification. Journal of Mathematical Psychology, 1970, 7, 293-316.

Norman, D.A. Personal Communication, June 1977.

Norman, M.F. Incremental learning on random trials. Journal of Mathematical Psychology, 1964, 2, 336-350.

Offir, J.D. Automaton models of performance. Journal of Mathematical Psychology, 1973, 10, 353-363.

Rasch, G. An Individualistic Approach to Item Analysis. In P.F. Lazarsfeld, and N.W. Henry (Eds.), Readings in Mathematical Social Science Chicago: Science Research Associates, 1966.

Smallwood, R.D. Optimal Policy Regions for Computer-Directed Teaching Systems. In W. Holtzman (Ed.), Computer-Assisted Instruction, Testing and Guidance. New York: Haper and Row, 1970.

Smallwood, R.D. The analysis of economic teaching strategies for a simple learning model. Journal of Mathematical Psychology, 1971, 8, 285-301.

Smallwood, R.D. Quantitative methods in computer-directed teaching systems. Stanford, California, Department of Engineering-Economic Systems, Stanford University 1967.

Trabasso, T., and Bower, G.H. Attention in Learning: Theory and Research. New York: Wiley, 1968.

Vruels, D., and Goldstein, I. In Pursuit of the Fateful Few: A Method for Developing Human Performance Measures for Training Control. Proceeding of the 7th NTEC/Industry Conference, 19-20 Novermber 1974. NAVTRAEQUIPCEN IH-240, 227-236.

Wickens, T.D., and Millward, R.B. Attribute elimination strategies for concept identification with practiced subjects. Journal of Mathematical Psychology, 1971, 8, 453-480.

Williams, G.E. A model of memory in concept learning. Cognitive Psychology, 1971, 2, 158-184.

Wollmer, R.D., Markov decision-model for computer aided instruction. Math Bioscience, 1976, 30, 213.

APPENDIX A

Section III contains three categories in which the different optimization techniques may be classified. These categories are listed by two concepts or terms which may be relatively unique to this report. These terms are; the unit-of-acquisition, and the unit-of-presentation. In order to illustrate these two concepts, consider the list of paired-associates shown in Figure 11.

d	NAL
DDD	RAP
ββ	TOG
BB	TOP
a	NIL
ΔΔΔ	RAG
aaa	RIL
Δ	NAG
AAA	RIP
ααα	RIG

Figure 11. Typical List of Paired-Associates

In paired-associate learning, one would be interested in requiring that the student eventually be able to respond correctly to all stimulus members in the list, and thus each pair is considered to be the unit-of-acquisition. In teaching the list, a standard paired-associate procedure would be to present the items one-at-time to the student, making several passes through the list, until the student is able to respond correctly to each stimulus. Thus the unit-of-presentation in this procedure is the same as the unit-of-acquisition. However, the list shown in Figure 11 was constructed according to the set of rules shown in Figure 12. As can be seen,

Key		Designation
First Letter	N = one T = two R = three	Number of Letters
Second Letter	I = A O = B A = D	Letters designated
Third Letter	P = Upper case L = Lower case G = Greek	Style of Letter

Figure 12. Rules by Which the First, Second and Third Response Letters are to be Generated as a Function of Stimulus Attributes

the characteristics of the response terms are dictated by the attributes of the stimulus members. For the first response of the list "NAL", the first letter was to be "N" because the stimulus contained only one letter. The second letter "A" was determined by the fact that the stimulus

contained one or more "ds" and the third letter "L" was determined by the fact that the stimulus letters were written in lower case. Thus if the student knew the rules contained in Figure 12, he could produce the composite "TAL" by attending to the attributes of the stimulus and using the rules to generate the response. In fact the student could respond correctly to the full range of paired-associate combinations as shown in Figure 13 without having

a	NIL
aa	TIL
aaa	RIL
A	NIP
AA	TIP
AAA	RIP
α	NIG
αα	TIG
ααα	RIG
b	NOL
bb	TOL
bbb	ROL
B	NOP
BB	TOP
BBB	ROP
β	NOG
ββ	TOG
βββ	ROG
d	NAL
dd	TAL
ddd	RAL
D	NAP
DD	TAP
DDD	RAP
Δ	NAG
ΔΔ	TAG
ΔΔΔ	RAG

Figure 13. Full Listing of Stimulus-Response Combinations Possible

been trained on each and every combination.

If the instructor was aware of the systematic relation between stimulus and response pairs, it would seem to be advantageous to train the student on the underlying rules rather than each individual S-R pair. In this situation, the units-of-presentation may still be considered to be the individual pairs, but the units-of-acquisition are now considered to simply be the underlying rules. In most instructional situations, the units-of-acquisition, like the present illustration, are fewer in number than the units-of-presentation.

As a further illustration, consider the stimulus members of our list to be flight-training exercises as shown in Figure 14. Here the

Exercises	Desired Responses
Descending right turn, constant velocity	TAL
Ascending right turn, decreasing velocity	RAP
Right turn, constant velocity altitude	TAG
Ascending right turn, constant velocity	TAP
Descending left turn, increasing velocity	NIL
Level right turn, decreasing velocity	RAG
Descending left turn, decreasing velocity	RIL
Level right turn, increasing velocity	NAG
Ascending left turn, decreasing velocity	RIP
Level left turn, decreasing velocity	RIG

Figure 14. Attributes of Exercise Requirements With Desired Response Requirements

attributes of the exercises have relationship to the responses similar to that found in the previous illustration. Consider also the possibility that the letter combinations found in the responses represent categories of multidimensional responses required in flight-training. Figure 15 shows the designated relationship. It will be noted that the rules shown are the same rules as in the previous illustration, only the definitions have changed. Thus the correct

Key	Response classes		Results
	categories	T	Constant velocity
First	of Throttle	N	Increased velocity
Letter	Action	R	Decreased velocity
	Responses	I	Left turn
Second	Required	O	Straight
Letter	in Turns	A	Right turn
	categories	P	Ascention
third	of stick	L	Descention
letter	Action	G	Level

Figure 15. Rules by Which Symbolic Responses are to be Generated as a Function of Exercise Attributes

response "TAL" indicates that the appropriate throttle response for constant velocity, turning responses, and stick movements be made for an exercise requiring a descending right turn with constant velocity. An incorrect response of say "NAL," would indicate that the student responded such that he failed to maintain constant velocity while making the descending right turn.

Several points should be noted in the illustrations just described. The first point is that the nature of most applied instructional situations is such that the units-of-presentation are fewer in number than the units-of-acquisition. The units-of-acquisition may be a set of concepts (or

underlying rules), or they may even be a set of basic psychomotor skills. Few real-life situations can be found where the instructional materials can be represented as a set of independent items as in the paired-associate analog. Usually the presentation of one unit affects the next unit to be presented . A practice trial on a descending left turn would obviously affect the student's chances of a successful completion of a descending right turn. In basketball, practice on free throws will have an affect on set shots. Thus it could be surmized that in these situations many different units-of-presentation could be used to affect learning on the few units-of-acquisition.

Figure 16 illustrates the idea that exercises affect learning

Underlying Concepts

Exercises	I	II	III
1	X	O	O
2	X	O	O
3	O	X	O
4	O	O	X
5	O	X	X
6	X	X	O
⋮			

Figure 16. Designation as to Which Exercises Affect the Learning of Particular Conceptual Rules

on one or more of the underlying units-of-acquisition which in turn affect subsequent performance on other exercises. It can be seen that exercise 1 affects only concept I (as indicated by the "x") while exercise six affects both concepts I and II. This illustration of the training characteristics of various exercises brings up a second important point about exercise selection in an adaptive logic. The point to be made is that instructional decisions should be based on inferred changes in the learning state of the units-of-acquisition. In looking at Figure 16, it can be seen that any decision as to the next exercise to present should be based on the inferred status of the student's learning of the three concepts rather than inferences of the status of each exercise. Thus it becomes apparent that a model of the student's learning process is needed. As an example, if it could be determined that concept I has a low probability of being in a learned state while concepts II and III have a high probability of being in a learned state, then any training system would probably limit its research for exercises to those like exercises 1 & 2. Further, the specific exercise selected should be the one with the highest likelihood of moving concept I into a learned state. Thus if a model could produce inferences as to the current learning states of the underlying concepts, exercise selection becomes more rational and less intuitive.

A third point concerns the principals of diagnostics and remediation. It will be recalled that the incorrect response of "NAL" (in place of "TAL") to the first exercise (descending right turn, constant velocity) indicated that the student was unable to maintain constant velocity. Thus while the response is categorized as being incorrect, the form of the incorrect response gives diagnostic information concerning the fact that the student has not yet acquired the first rule listed in Figure 15. The learning model may in fact use this information to update its inferences on the status of the different conceptual rules. Thus remediation becomes a matter of simply selecting an appropriate exercise.

The remainder of section III contains specific optimization techniques which were found in the literature. The techniques are grouped as to whether they pertain to training situations wherein; the units-of-acquisition are the same as the units-of-presentation (the paired-associate analog), the units-of-acquisition are concepts (rules or highly cognitive rule-oriented psychomotor tasks), or where the units-of-acquisition pertain to simple motor skills.

DISTRIBUTION LIST

Naval Training Equipment Center
Orlando, FL 32813 — 33

Defense Documentation Center
Cameron Station
Alexandria, VA 22314 — 12

Headquarters
Air Training Command, XPT
Attn: Dr. John Meyer
Randolph AFB, TX 78148 — 2

All other addressees receive one copy.

Seville Research Corp.
Suite 400 Plaza Bldg
Pace Blvd at Fairfield
Pensacola, FL 32505

USAHEL/AVSCOM
Dir, RD&E
ATTN: DRXHE-AV (Dr. Hofmann)
P. O. Box 209
St Louis, MO 63166

Army Training Support Center
Ft Eustis, VA 23604

Commandant
USA Field Artillery School
Target Acquisition Dept
ATTN: Eugene C. Rogers
Ft Sill, OK 73503

Director
Human Resources Research Organization
300 N Washington St
Alexandria, VA 22314

Human Resources Research Organization
Division No. 1, Systems Operation
300 N Washington St.
Alexandria, VA 22314

Chief of Naval Research
Code 458, Dept of Navy
Arlington, VA 22217

Chief of Naval Research
Psychological Sciences
Code 455, Dept of Navy
Arlington, VA 22217

Chief of Naval Operations
OP-991B, Dept of Navy
ATTN: M. K. Malehorn
Washington, DC 20350

Chief of Naval Operations
OP-987H, Dept of Navy
ATTN: Dr. R. G. Smith
Washington, DC 20350

Library
Navy Personnel Research and Development Center
San Diego, CA 92152

Grumman Aerospace Corp
Plant 36
ATTN: Mr. Sam Campbell
Bethpage, LI, NY 11714

Texas Technical University
Psychology Dept, Box 4100
ATTN: Dr. Charles Holcomb
Lubbock, TX 79409

Director
Defense Research and Engineering
ATTN: LCOL H. Taylor, OAD E&LS
Washington, DC 20301

CDR J. E. Goodson, MSC, USN
Aerospace Psychology Dept. (L53)
Naval Aero Medical Rsch Lab
Naval Air Station
Pensacola, FL 32508

Commander
Naval Sea Systems Command
Code 03416 (Mr. P. J. Andrews)
Washington, DC 20360

Commander
Naval Air Development Center
ATTN: Training Branch (6043)
Warminster, PA 18974

Human Factors Engineering
Division
NAVAIRDEVCEN, Code 4024
ATTN: LCDR Charles Theisen
Warminster, PA 18974

Commanding Officer
PAC MISS TEST CTR
ATTN: Hd Human Factors,
Engineering Branch
Pt Mugu, CA 93042

Naval Technical Training
Command (Code 0161)
NAS Memphis (75)
Millington, TN 38054

Chief of Naval Technical
Training (Code 0161)
NAS Memphis (75)
Millington, TN 38054

Commanding Officer
NAVED TRASUPPCENPAC
Code N5B
ATTN: Mr. Rothenberg
San Diego, CA 92132

US Air Force Human Resources Lab
AFHRL/OR Occupational Manpower
Relations Div
Lackland AFB, TX 78235

US Air Force Human Resources Lab
AFHRL/SMS
Computational Sciences Division
Statistical & Computer
Technology Branch
Brooks AFB, TX 78235

US Air Force Human Resources
Lab/DOJZ
Brooks AFB, TX 78235

AFHRL/FTO
ATTN: Mr. R. E. Coward
Luke AFB, AZ 85309

US Air Force Human Resources Lab
AFHRL-TT
Technical Training Division
Lowry AFB, CO 80230

US Air Force Human Resources Lab
AFHRL-FT
Flying Training Division
Williams AFB, AZ 85224

ASD SD24E
ATTN: Mr. Harold Kottmann
Wright-Patterson AFB, OH 45433

ASD/ENETC (Mr. R. G. Cameron)
Wright-Patterson AFB, OH 45433

Commander
Navy Air Force, US Pacific Fleet
NAS North Island (Code 316)
San Diego, CA 92135

Commander
Training Command
ATTN: Educational Advisor
US Pacific Fleet
San Diego, CA 92147

AFHRL/PE
Brooks AFB, TX 78235

Chief
ARI Field Unit
P. O. Box 2086
Fort Benning, GA 31905

Chief
ARI Field Unit
P. O. Box 476
Fort Rucker, AL 36362

Chief
Naval Education & Training
Liaison Office
AF Human Resources Laboratory
Flying Training Div
Williams AFB, AZ 85224

Commander
Naval Air Systems Command
Naval Air Systems Command HQS
(AIR 340F)
Washington, DC 20361

Commander
Naval Air Systems Command
Naval Air Systems Command HQS
(AIR 413-B)
Washington, DC 20361

Naval Weapons Center
Code 3143
ATTN: Mr. George Healey
China Lake, CA 93555

Chief of Naval Education and
Training Liaison Office
Human Resource Laboratory
Flying Training Div
(ATTN: CAPT W. C. Mercer)
Williams AFB, AZ 85224

TAWC/TN
Eglin AFB, FL 32542

HQ ADCOM/DOXI
Peterson AFB, CO 80914

Commandant
US Army Field Artillery School
ATSF-TD-TE (Mr. Inman)
Ft Sill, OK 73503

Technical Library
US Army Concepts Analysis
Agency
8120 Woodmont Ave
Bethesda, MD 20014

Scientific Advisor
HQ US Marine Corps
Washington, DC 20380

Calspan Corp.
Librarian
PO Box 235
Buffalo, NY 14221

Commanding Officer
Code 104, Bldg S-54
NATTC, NAS Memphis (85)
Millington, TN 38054

AFHRL/AS
Wright-Patterson AFB, OH 45433

Headquarters, ESD/DRI
Hanscom AFB, MA 01731